



Adaptive sequencing using nanopores and deep learning of mitochondrial DNA

Artem Danilevsky , Avital Luba Polsky and Noam Shomron 

Corresponding author: Noam Shomron, Faculty of Medicine and Edmond J Safra Center for Bioinformatics, Tel Aviv University, Tel Aviv 69978, Israel.
Tel.: +972-3-640-7387; Fax: +972-3-640-7432; E-mail: nshomron@tauex.tau.ac.il

Abstract

Nanopore sequencing is an emerging technology that reads DNA by utilizing a unique method of detecting nucleic acid sequences and identifies the various chemical modifications they carry. Deep learning has increased in popularity as a useful technique to solve many complex computational tasks. ‘Adaptive sequencing’ is an implementation of selective sequencing, intended for use on the nanopore sequencing platform. In this study, we demonstrated an alternative method of software-based selective sequencing that is performed in real time by combining nanopore sequencing and deep learning. Our results showed the feasibility of using deep learning for classifying signals from only the first 200 nucleotides in a raw nanopore sequencing signal format. This was further demonstrated by comparing the accuracy of our deep learning classification model across data from several human cell lines and other eukaryotic organisms. We used custom deep learning models and a script that utilizes a ‘Read Until’ framework to target mitochondrial molecules in real time from a human cell line sample. This achieved a significant separation and enrichment ability of 2.3-fold. In a series of very short sequencing experiments (10, 30 and 120 min), we identified genomic and mitochondrial reads with accuracy above 90%, although mitochondrial DNA comprised only 0.1% of the total input material. The uniqueness of our method is the ability to distinguish two groups of DNA even without a labeled reference. This contrasts with studies that required a well-defined reference, whether of a DNA sequence or of another type of representation. Additionally, our method showed higher correlation to the theoretically possible enrichment factor, compared with other published methods. We believe that our results will lay the foundation for rapid and selective sequencing using nanopore technology and will pave the approach for clinical applications that use nanopore sequencing data.

Keywords: nanopore-sequencing, selective-sequencing, deep-learning, real-time, classification

Introduction

Next-generation sequencing

Next-generation sequencing has revolutionized DNA sequencing and laid the foundation for a plethora of scientific and clinical opportunities. One recent emerging sequencing technology uses nanopore sequencing (e.g. those developed by Oxford Nanopore Technologies, ONT) [1]. In this study, we used ONT’s portable MinION sequencer, which was released in 2014. The sequencing is performed by measuring changes in ionic current produced by individual nucleic acids as single DNA strands that pass through an array of protein nanopores. These changes are detected by a sensor and are saved on a computer for later analysis [2]. The recorded ionic current, known as the ‘raw signal’ or ‘squiggle’, is mainly used for basecalling by translating the raw signal into nucleotides. To date, the vast majority of studies that used nanopore sequencers ignored the raw signal after using it to generate a nucleotide sequence. A few studies, however, used this signal for other tasks, such as

improving the accuracy of a consensus sequence, or for investigating chemical modifications on the DNA [3–5].

Deep learning

Deep learning is a subset of machine learning methods that have gained increased popularity in recent years, after overtaking other methods in the field of image classification [6]. Deep learning has been applied to fields such as image, video, audio and natural language processing, for performing tasks such as classification, generation, prediction and detection [6–8]. This raises the possibility that similar deep learning approaches can be applied to nanopore sequencing data analysis. These approaches include methods that have been used for audio signal analysis, such as convolutional neural network (CNN) and recurrent neural network (RNN) architectures [6, 9–11]. Initially, raw nanopore signals were translated to nucleotides using a Hidden Markov Model [12, 13]. However, the discovery that deep learning can

Artem Danilevsky completed his undergraduate and master’s degrees at Tel Aviv University and is currently working toward his PhD in medical sciences at Tel Aviv University.

Avital Luba Polsky completed her undergraduate degree at New Mexico State University, and her graduate degree at the Technion – Israel Institute of Technology. **Noam Shomron** is a Full Professor at Tel Aviv University and the Head of the Functional Genomics Laboratory at Tel Aviv University’s Medical School, as well as the Digital Medicine Laboratory, as part of the Tel Aviv University Innovation Labs (TILabs). He is also the Director of Djerassi Institute of Oncology and Rare Genomics Institute Israel.

Received: November 16, 2021. **Revised:** May 13, 2022. **Accepted:** May 30, 2022

© The Author(s) 2022. Published by Oxford University Press. All rights reserved. For Permissions, please email: journals.permissions@oup.com

perform the task better led to its use in translating a raw nanopore signal into a nucleotide sequence [14, 15]. Deep learning is also used to perform such tasks as predicting DNA methylation [16] and simulating a raw signal based on a reference genome [17]. These findings support our proposition that deep learning can be used to classify reads based on their raw signal. Hence, we tested several commonly used deep learning architectures that were previously applied on similar data, with the aim of selecting the preferable one for our analysis.

Traditional selective sequencing (targeted sequencing)

Selective sequencing (or sequencing of targeted genomic regions) is a widespread technique used in many applications designed to sequence specific portions of a DNA molecule from a larger pool of genetic material. The targeting of only a part of the DNA can save resources, time and money. Selective sequencing is traditionally based on physically isolating or amplifying desired parts of the total DNA during the library preparation steps and prior to sequencing [18–20]. Recently, this approach was also implemented during standard nanopore library preparation [21, 22]. Traditional selective methods, however, have been found to introduce bias to the output, such as lack of evenness of coverage and divergent results from different library preparation kits [23]. Additionally, all current targeted sequencing techniques rely on amplifying a region of interest that is based solely on the nucleotide sequence. This prompts the need for an alternative method.

Nanopore adaptive sequencing

With the introduction of nanopore sequencing, an exciting new feature, ‘Read Until’, enables selectively ‘rejecting’ DNA molecules before the entire molecule has been completely sequenced [24]. The rejection of molecules based on the initial portion of the DNA molecule potentially saves time and reagents by sequencing only selected DNA molecules. This establishes a unique method of selective sequencing on the nanopore sequencing platform. Several studies have demonstrated real-time selective sequencing using the nanopore Read Until feature. Notably, in a first published study, Loose *et al.* [24] demonstrated the ability to perform selective sequencing with the genome of Lambda phage; dynamic time warping was used to determine the DNA molecules to be sequenced. This approach stipulated the length of the possible target and reference sequences. In another study, Edwards *et al.* [25] performed real-time selective sequencing by online basecalling the start of the molecules and then selecting the molecules to sequence by mapping them to a reference library using the LAST aligner. Similarly, Payne *et al.* [26] mapped the base-called nucleotides to a reference genome using minimap2. This approach removed the constraints imposed by the dynamic time warping algorithm; however, it introduced two distinct steps (basecalling and mapping) into the

decision process. Another study used the same concept of basecalling and mapping the reads to a reference genome, but for a different purpose, namely, to achieve more uniform coverage [27]. Finally, in a more recent work, Kovaka *et al.* probabilistically decoded the raw signal into k-mers that could be represented by the signal, and later used to align the signal to a reference. They compared this k-mer composition to a k-mer representation of a known reference. This led to an enrichment of 4.46 [28].

Overall, the above methods, de facto, comprised processing the raw signal and subsequently comparing it to a known reference. Our unique method enables classification of a raw signal without direct reliance on a reference or prior knowledge at the time of decision making (classification of the raw signal from the DNA molecule currently at the nanopore). The method only requires using the trained model for classification, which learns the intrinsic patterns of each group of different reads.

Our contribution

Here we apply selective sequencing on nanopore sequencing via a unique deep learning approach. We began by developing a deep learning model capable of accepting only the first 2000 values of a raw signal, which equates to roughly 200 base pairs as input. We chose to perform selective sequencing on mitochondrial DNA, which is a cellular organelle within eukaryotic cells, containing about 16 K base pairs and encoding 13 proteins. Mitochondrial DNA has been sequenced many times, has high coverage in publicly available nanopore datasets and is of biological and medical significance in the analysis of human sequencing data [29]. We trained the model to classify sequencing reads as ‘mitochondrial’ or ‘genomic’ based on the signal. Analysis of the raw signal directly bypasses the basecalling step, which is potentially error prone; while enabling the deep learning model to incorporate additional information present in the raw signal, such as DNA modifications [4, 5]. Although this experiment relied on information produced by the basecalling process, the classification models can likely be trained without basecalling at all, but rather from two groups on nanopore signals, separated by methods other than mapping. This approach can be further improved, and eventually surpass the current basecalling models. The upshot is potentially increased accuracy of DNA classification by eliminating data analysis steps and increasing the information volume for the deep learning model. Unlike previous attempts at real-time selective sequencing, our method requires neither a nucleotide reference nor a generated signal reference at the time of sequencing. Bypassing a reference decreases the run time and complexity restriction as the reference database expands. This is because in using other reference-based approaches, searching and parsing the data add size and complexity challenges. Moreover, our model can potentially use information that is not used

by simple mapping algorithms, such as methylation and nucleotide composition. This can facilitate the rapid differentiation of complex DNA samples that are limited in their detection capacity without complete processing (mapping and referencing). The method might pose an advantage for clinical applications. We tested several deep learning architectures for sequence analysis and used several datasets of nanopore signal data, and applied them to classifying reads of a number of DNA origins. We selected the model with the highest classification accuracy and combined it with the Read Until API. The goal was to perform a sequencing experiment that used our model to successfully select and sequence mitochondrial DNA. The general workflow of this method is illustrated in Figure 1. Finally, we compared our results to theoretically expected results, based on the official information provided by ONT [30]. Our model achieved the expected enrichment results considering our experimental parameters.

Overall, the development of a new real-time selective sequencing method will not only alleviate the challenges posed by the additional steps during library preparation for targeted sequencing, but will enable classification of entire DNA molecules based on the raw signal. The latter is impossible by the alternative adaptive sequencing method of translation and mapping signals. Our method has the potential to increase accuracy and to accelerate the sequencing process, and can eventually be applied to any clinical setting in which time-sensitive DNA sequencing is of the essence.

Methods

Data organization, preprocessing and augmentation

For the purpose of training and testing our deep learning models, we used two publicly available nanopore sequencing datasets: a dataset established by Jain *et al.* [31] and the 'Cliveome' dataset. Jain *et al.* [31] produced a human genome assembly using long reads from nanopore sequencing. About 14 million reads were sequenced and aligned to the 1000 genome GRCh38 reference genome [32]. From this dataset, we used 60 000 reads that were aligned to the mitochondria and 200 000 random reads that were aligned to the rest of the human genome. The 'Cliveome' dataset was sequenced by ONT and released to the public in 2016 [33]. From this dataset, we used 8000 reads that mapped to the mitochondria, and 200 000 random reads that mapped to the rest of the human genome. In each dataset, we separated the sequenced reads randomly into training, validation and test sets containing 80%, 10% and 10%, respectively, of the total reads. Only the first portion of each raw signal was used to simulate reading the beginning of the molecule with the Read Until feature.

Deep learning requires iterating through the training dataset by mini-batches. This enables handling large datasets and improves the training results [34]. In this

research we used the Pytorch [35] deep learning framework, which contains a Dataloader class; we customized this class to allow parallel data loading with custom data transformations. Our custom dataloader applies four transformations to the signal. The first transformation randomly selects a region of 2000 values from the total 5000 values. The second transformation changes the signals from the raw values, which represent the electric current level, to differential values. This eliminates possible bias between voltages of different devices and flow cells. The third transformation cuts the signal into a sliding window array, transforming the 1D-long linear signal into a 2D array of stacked sliding windows. The final transformation adds Gaussian noise to the sample, to mimic the background noise in nanopore sequencing. All the transformations improved the training process and the final accuracy; further details are provided in the Supplementary Methods (see Supplementary Data available online at <http://bib.oxfordjournals.org/>).

To assess the applicability of our classification model to other datasets, nanopore sequencing data from a number of cell lines and organisms were gathered from the publicly available SRA databases. A total of 13 samples were gathered, five samples of human cell lines of various origins and eight samples from other organisms. The samples and their corresponding identifiers are: Human-Liver [ERR4395307], Human-Colon [SRR12880625], Human-Tongue [SRR13920755], Human-B-lymphocyte-1 [SRR10359548], Human-B-lymphocyte-2 [SRR13920753], Primate-Macaque [SRR12517384], Bird-Macaw [ERR6797213], Fish-Tilapia [SRR11744846], Sponge [ERR3619183], Nematode [SRR11565827], Fly [SRR8627922], Fungi [SRR9690733] and Potato [SRR10489253].

Model architecture, training and testing

We tested five neural network varieties for our deep learning model architecture: regular CNN [36], very deep CNN (VDCNN) [37], regular long short-term memory (LSTM) [38], LSTM with recurrent batch normalization [39] and regular gated recurrent unit (GRU) [40]. Additional details of the models and the rationale for their selection are presented in the Supplementary Methods (see Supplementary Data available online at <http://bib.oxfordjournals.org/>). All the models were tested as three sizes, corresponding to the number of hidden parameters: large, medium and small. All the models were tested extensively with various configurations, as explained in the Supplementary Methods (see Supplementary Data available online at <http://bib.oxfordjournals.org/>).

We also attempted to combine a CNN model with an RNN model, as illustrated in Figure 1. In theory, CNN is good at proximal feature representation and RNN can find long distance dependencies. By combining these techniques, our model utilizes both short- and long-distance information hidden in the raw signal [41]. We combined the VDCNN with regular GRU,

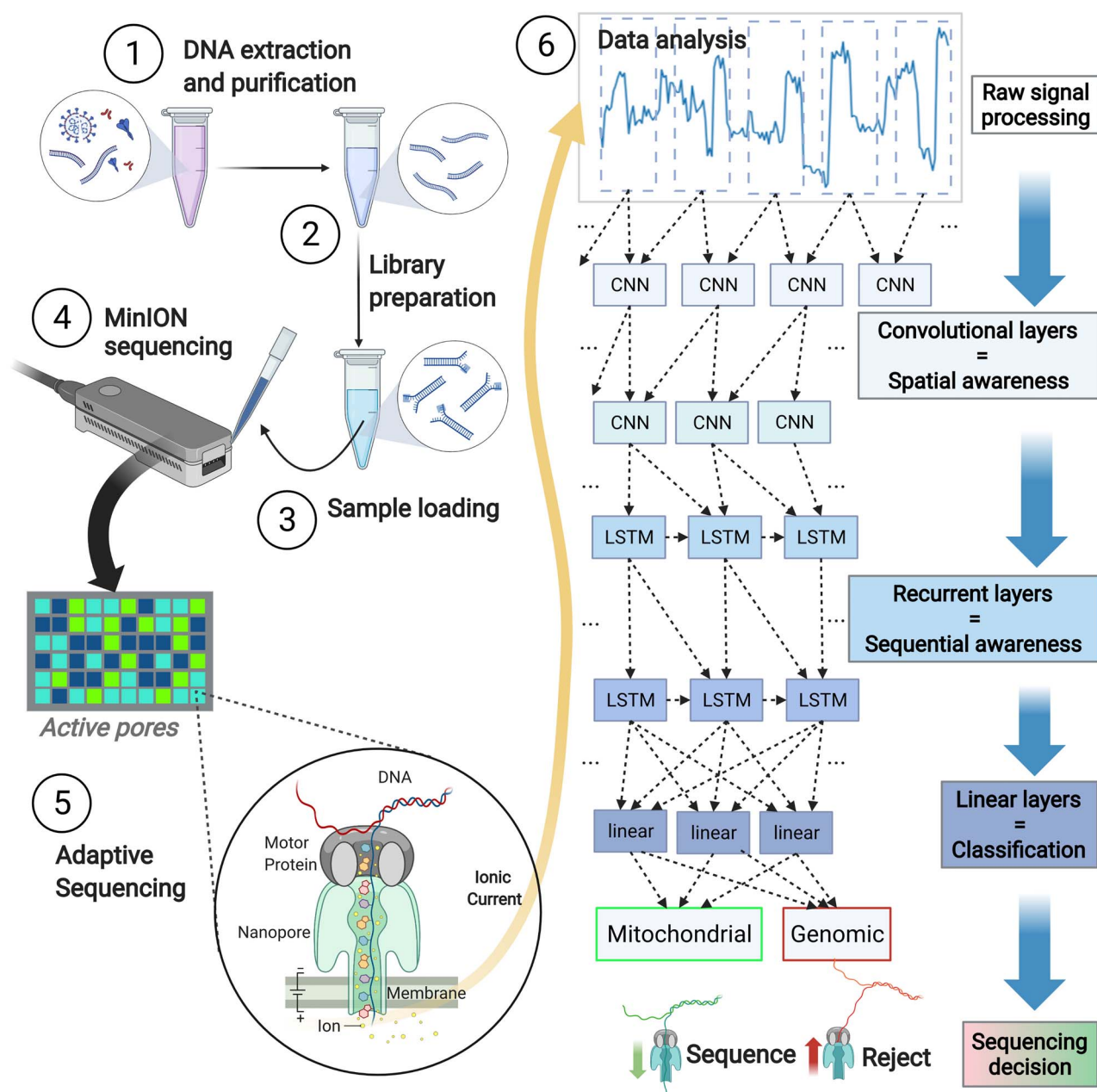


Figure 1. An illustration of the sequencing workflow with an expanded schematic overview of the information flow in the deep learning model, which combines the CNN and RNN-type layers, from the signal to a final classification.

and VDCNN with LSTM. We applied recurrent batch normalization and tested multiple configurations of these models, as described in the Supplementary Methods (see Supplementary Data available online at <http://bib.oxfordjournals.org/>). In total, 90 configurations of models with various parameters and architectures were tested. Unlike the parameters of the model which directly impact its structure and therefore performance, hyperparameters (such as learning rate, batch size and regularization) control the process of training and have negligible impact on the final performance of the model. During training, the hyperparameters were either kept at their default settings or manually adjusted to allow more efficient training without impacting the final results.

Notably, the default values were selected based on successful training experiments in previous publications [6, 34].

To mitigate for discrepancies in model accuracy due to different training processes, the same python script was used to train all the models similarly. To assess the accuracy of each model, we used the training dataset during training, the validation dataset for hyperparameter tuning and the test dataset exclusively at the final stage. Accuracy was assessed separately for genomic reads and mitochondrial reads. Total accuracy was calculated by averaging the accuracy of the mitochondrial and the genomic reads. An Adam (A Method for Stochastic Optimization) optimizer [42]

was used; the learning rate and other parameters for the optimizer were determined by a manual search. All the models were trained for 300 epochs, and the learning curve of each model was assessed to examine possible plateauing of the loss curve and overfitting. [Supplementary Figure 1](#) (see Supplementary Data available online at <http://bib.oxfordjournals.org/>) illustrates the learning curve of the model with LSTM and recurrent batch normalization, as an example of a successfully trained model.

After training all the models on the primary dataset, a second dataset (Cliveome) was used to test the models for generalization. At first, the accuracy of the models was tested on the test dataset from the second dataset, without any additional training. Later, all the models were trained for 30 epochs on the second dataset training data, to improve accuracy specifically for the second dataset. This approach is termed finetuning [43], which allows to initially train a model on a vast dataset and later perform minimal amount of additional training on a different data in order to increase the performance of the model on the new data. After the additional training, all the models were tested again with the second dataset, and the accuracy was recorded.

To assess the ability of the classification model to classify samples from different sources, we gathered 13 samples of raw nanopore sequencing data from various datasets. Each sample was basecalled using the Guppy basecaller (version 5.0.16), and aligned to the appropriate genome. Individual sequencing reads were separated into two groups based on their alignment: mitochondrial reads and genomic reads. The same number of reads from each group was used for the analysis and up to 1000 random reads from each group were selected for the analysis (per sample). Even though the proportion of mitochondrial and genomic reads might be different in every experiment, analyzing a dataset where the proportion of mitochondrial and genomic reads are 1:1 provides a more accurate performance measurement and allows us to spot biases, if present, more easily. The best performing model that was previously trained on the primary and secondary datasets was used to classify the reads and to calculate accuracy. Evolutionary distance scores were calculated for each sample (other than human cell lines), based on the TimeTree [44] database as the distance between humans and the source organism of the sample. The correlation between accuracy and the evolutionary distance was calculated using Pearson's correlation coefficient.

DNA extraction, library preparation and MinION sequencing

Monolayer-adherent HEK-293T cells (transformed human embryonic kidney cells, ATCC, USA) were grown in Dulbecco's modified Eagle's medium (DMEM) (Thermo Fisher Scientific, USA) supplemented with 10% (vol/vol) fetal bovine serum (FBS) (Thermo Fisher Scientific, USA), 0.3 g/liter L-glutamine, 100 unit/ml penicillin and

100 units/ml streptomycin (Biological Industries, Israel). Cells were incubated at 37°C in 5% CO₂ atmosphere. Before use, cells were confirmed to have no mycoplasma contamination using the EZ-PCR Mycoplasma Test Kit (Biological Industries, Israel). Prior to each experiment, the cells were counted using the Countess automated cell counter (Thermo Fisher Scientific, USA).

Qiagen's QIAamp DNA mini kit was used to extract DNA from HEK-293T cells. Next, 2.5×10^6 or 1×10^6 cells were centrifuged at 1400×g for 5 min, and the resulting pellet was resuspended in 200 µl PBS. DNA was then extracted according to the manufacturer's protocol and eluted in 200 µl H₂O. The DNA concentration was measured using the dsDNA High Sensitivity Assay on a Qubit fluorometer (Thermo Fisher Scientific, USA). DNA purity was assessed using a NanoDrop ND-1000 spectrophotometer (NanoDrop Technologies, Thermo Scientific, USA), to ensure OD 260/280 and OD 263/230 > 1.8.

About 400 ng of purified DNA in a total volume of 7.5 µl in a 0.2 ml PCR tube was used as input for sequencing library preparation using ONT's Rapid Sequencing kit (SQK-RAD004, version RSE_9046_v1_revB_17Nov2017) according to the manufacturer's instructions. For fragmentation and transposase adapter attachment, 2.5 µl FRA was added to the DNA and mixed by inversion. The sample was then incubated at 30°C for 1 min, followed by 80°C for 1 min, and finally cooled on ice. Sequencing adapters were then attached by adding 1 µl RAP to the mixture and mixing by inversion. The sample with sequencing adapters was incubated at room temperature for 5 min, and then stored on ice until it was sequenced.

MinION sequencing was conducted according to the manufacturer's instructions using R9.4 and R9.4.1 rev. D flow cells (FLO-MIN106, ONT). After flow cell priming, 4.5 µl nuclease-free water, 34 µl sequencing buffer and 25.5 µl mixed loading beads were added to the library and mixed by gently flicking the tube immediately before loading into the SpotOn port.

Three sequencing experiments were performed under those conditions; they will be referred to as the 'HEK1', 'HEK2' and 'HEK3' runs. The results of the first two experiments were used for testing and training the model, whereas the third experiment was run in conjunction with the Read Until script to perform real-time selective sequencing.

Sequencing data analysis

After data were acquired from the first two sequencing experiments, HEK1 and HEK2, the reads were translated to nucleotides using ONT Albacore version 2.2.5. Although Albacore is an older version of the official basecaller, a recent comparison between basecalling software indicated that its differences from the more modern basecallers [45] are negligible regarding aspects relevant to our experiments. The reads were mapped to the GRCh38 human reference genome using minimap2 software [46] version 2.11. Reads were stratified as mitochondrial or genomic based on their mapping, and each

group was separated into training/validation/testing groups with proportions of 80%/10%/10% of the total reads, respectively. Initially, the accuracy of the models trained on the first dataset was tested with the HEK1 data. Later, the models were trained for 30 epochs on the HEK1 data and accuracy was tested again (finetuned). The best performing model was determined by the highest accuracy value on the HEK2 data, and saved for later use with Read Until on the HEK3 sequencing experiment.

To test the performance of Read Until, we utilized the developmental API provided by ONT, and wrote a custom script to perform selective sequencing based on the 'simple.py' file from the GitHub repository of Read Until. This script receives the raw signal at the beginning of every DNA molecule. The signal is analyzed by the deep learning model; and the script eventually sends a signal to the MinION device to continue sequencing the DNA molecule or to stop and remove the DNA molecule from the pore. Reads that are classified by the model as mitochondrial are fully sequenced, whereas the remaining reads receive a signal to terminate their sequencing. To attain validated results, we performed the experiments with three repeats for three time spans: 10, 30 and 120 min. In each time span we performed three regular sequencing experiments without using Read Until, and three sequencing experiments utilizing Read Until. To account for the deterioration of the flow cell over time and to mitigate technical bias, we performed the experiments without Read Until and while using it sparingly. We used the same flow cell during all the runs of the adaptive sequencing experiment. This enabled minimizing technical bias, which could occur from signals arriving from different flow cells; and eliminating bias originating from differences in available pore numbers. In contrast to other researchers, we intentionally did not separate the pores and run adaptive sequencing on groups of reads, as we noted reports of uneven deterioration and variability between the selective sequencing pores in some of the works. The reads were translated and mapped to a human reference genome. For each sequencing experiment, the alignment statistics were collected. Logistic regression with proportions and a random effects variable [47] analysis were performed to examine differences in the proportion of the sequenced mitochondrial nucleotides and in the total sequenced nucleotides. Pairs of the technical repeats were compared as follows: 10min_with_read-until_run1 versus 10min_without_read-until_run1, 10min_with_read-until_run2 versus 10min_without_read-until_run2, etc. Additionally, to examine statistical differences between the read lengths of different groups, read lengths were collected for each of the experiments and analyzed using the Fisher-Pitman permutation test [48], the calculations considered the paired information, when applicable (measurements within the same experiment).

To compare the adaptive sequencing performance of our deep learning model to other published methods,

performance metrics of five previous studies mentioned in section 1.4 of the Introduction were gathered and analyzed. Experimental enrichment values were collected from the publications 'as is', if they were reported; and they were calculated based on the experimental data, if they were not reported. The 'experimental' to 'theoretical' ratio values were calculated using the appropriate sequencing parameters, as described in Discussion section. The ratio of enrichment was calculated as the quotient of the 'experimental' enrichment and the 'theoretical' enrichment.

Results

Deep learning model selection

We trained 90 configurations of models in total while saving the accuracy statistics (see Supplementary Data File 1, see Supplementary Data available online at <http://bib.oxfordjournals.org/>). More than half the models exhibited total accuracy above 70% for all datasets after training. Table 1 summarizes the results. In general, larger models achieved higher accuracy than smaller versions, as can be seen in Supplementary Data File 1, see Supplementary Data available online at <http://bib.oxfordjournals.org/>. In addition, the models performed better after finetuning on a particular dataset (Table 1 and the remaining results in Supplementary Data File 1, see Supplementary Data available online at <http://bib.oxfordjournals.org/>). Furthermore, the addition of a dropout or a batch normalization layer generally improved performance in all the models. Comparing architecture types showed that the RNN type models—regular LSTM, LSTM with regular batch normalization, and GRU, achieved higher accuracy than the CNN-type networks—regular CNN and VDCNN. Table 1 presents the total accuracy scores.

Classification ability of samples from different sources

The LSTM that performed best with the recurrent batch normalization model was used to classify raw nanopore sequencing reads of 13 samples from various sources (Figure 2). Classification accuracy was highest for samples originating from human cell lines, in the range of 80–90%. Classification accuracy for samples originating from other organisms was negatively correlated with the evolutionary distance score. Accordingly, as the evolutionary score of the source organism of the sample increased, the accuracy of prediction decreased (Pearson correlation r [11]: -0.89 , P -value: 2.97×10^{-5}).

Real-time selective sequencing with Read Until

Based on the above results, we selected the LSTM with the recurrent batch normalization model that achieved the highest accuracy, 95.81%, with the HEK 2 data; and the highest accuracy overall, 92.45%. This model was used in conjunction with Read Until to perform real-time

Table 1. Accuracy values of the best performing deep learning models as assessed on various datasets

	Data from public datasets			Experimental data				
	Primary dataset (NA12878)	Secondary dataset (ONT), without finetuning	Secondary dataset (ONT), with fine-tuning	HEK1, with-out fine-tuning	HEK1, with fine-tuning	HEK2, with-out fine-tuning	HEK2, with fine-tuning on HEK1	Total accuracy
CNN with 3 layers, ^{+D,Small}	56.61%	57.20%	66.30%	62.48%	63.10%	77.38%	55.00%	62.58%
VDCNN, ^{+MP,+S,Medium}	88.70%	72.22%	72.83%	93.83%	77.67%	82.56%	74.79%	80.37%
LSTM, ^{+BN,Large}	82.40%	68.14%	73.61%	86.62%	86.26%	81.81%	81.73%	80.08%
LSTM with recurrent BN, ^{+D,+LS,Large}	89.42%	82.98%	87.52%	98.01%	97.93%	95.48%	95.81%	92.45%
GRU, ^{+BN,+HO,Large}	94.04%	81.72%	83.64%	98.61%	94.55%	96.13%	93.60%	91.76%
VDCNN + LSTM with recurrent BN, ^{+BN,+D,+HO, Large}	94.71%	63.07%	89.40%	96.45%	97.60%	80.33%	91.70%	87.90%
VDCNN + GRU, ^{+D,+LS,Large}	88.37%	60.76%	77.94%	92.40%	95.39%	87.29%	84.20%	83.76%

Small/Medium/Large, refers to the size of the model; +D, dropout; +S, shortcut; +MP, max-pooling; +BN, regular batch normalization; +LS, last step taken from RNN; +HO, hidden output taken from RNN; ONT, Oxford nanopore technologies; HEK, human embryonic kidney cells; CNN, convolutional neural network; VDCNN, very deep CNN; LSTM, long short-term memory; GRU, gated recurrent unit.

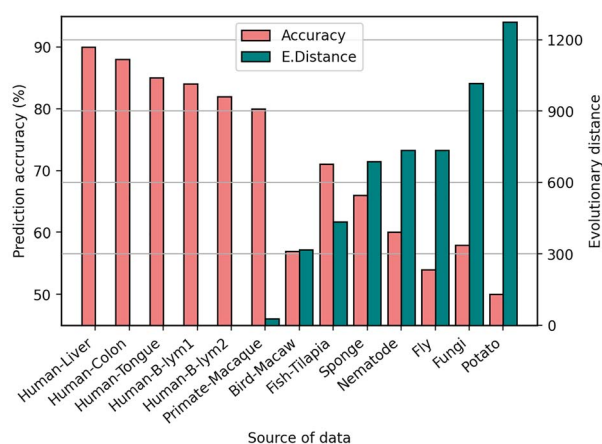


Figure 2. Classification accuracy of mitochondrial and genomic nanopore signals from various organisms (pink bars), together with the evolutionary distance of each organism from a human reference (green bars). The samples are ordered by their evolutionary distance score.

selective sequencing. The Read Until script was configured to sequence only molecules that were classified as mitochondrial reads by the model. During sequencing HEK3, the accuracy of the model was above 90%, which corresponds to the accuracy measured on HEK1 and HEK2 without finetuning.

The enrichment factor of our method was measured by calculating differences in the percentage of mitochondrial nucleotides between experiments, with and without selective sequencing. This was carried out to normalize the samples for total sequencing output and to eliminate any variance due to interchangeability during the experiments. We achieved a normalized enrichment factor of 2.3X (P -value < 0.05 , Figure 3), which was calculated using logistic regression with proportions and a random effects variable, as was best suited for statistical comparison of proportions. Comparing the means of the mitochondrial nucleotides in all the

experiments, with and without selective sequencing (normalized or not), we achieved an enrichment factor of 1.34X. Though most of the molecules were classified as genomic by the deep learning model and should not have been sequenced, most of the molecules classified as genomic were sequenced and saved to the hard-drive (see the Discussion).

Table 2 summarizes the results of all the selective sequencing experiments. The mean percentage of mitochondrial nucleotides for each time interval is shown, averaged across three experiments performed for each time interval. In addition, the average read length is shown for all the reads. The percentage of mitochondrial nucleotides differed significantly between sequencing experiments with selective sequencing and those without selective sequencing (P -value < 0.05).

In addition to differences in the percentage of nucleotides, read lengths differed distinctly between the groups. In examining lengths of the mitochondrial and genomic reads in experiments without selective sequencing, and considering the pairs of read lengths within each experiment, no significant difference was found (P -value > 0.8 , difference between means ≈ 75). However, there was a trend toward longer mitochondrial than genomic reads in experiments with selective sequencing (P -value < 0.1 , diff ≈ 1400). Mitochondrial reads were shorter in experiments with than without selective sequencing (P -value < 0.005 , diff ≈ 2100). Larger and more statistically significant differences were observed between the genomic read lengths in experiments with selective sequencing than in those without it; the genomic read lengths were significantly longer in experiments without selective sequencing (P -value < 0.0005 , diff ≈ 3600).

Table 3 compares previously published adaptive sequencing methods. Experimental enrichment provides a simple metric of the absolute enrichment for the targeted DNA region in the sample, and increased

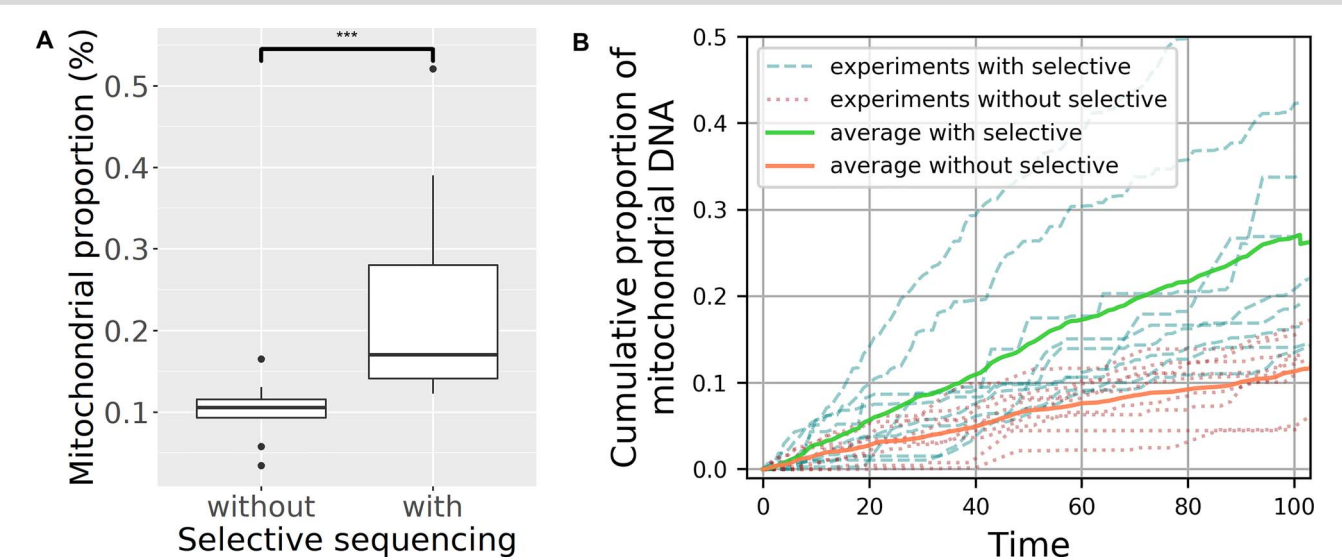


Figure 3. Differences in the percentages of mitochondrial nucleotides between experiments with and without selective sequencing. **(A)** Box plot illustrating differences in the mitochondrial reads between experiments with and without selective sequencing. **(B)** Cumulative percentage of mitochondrial DNA in relation to the final amount of total DNA throughout the experiments. The timeframes are adjusted to a scale of 0–100% of the experiment’s duration. Green and red solid lines denote the mean percentages of mitochondrial DNA throughout all the experiments, with and without selective sequencing, respectively. Light green and light red dotted lines denote the percentages of mitochondrial DNA throughout individual experiments, with and without selective sequencing, respectively.

Table 2. The mean coverage of mitochondria and the mean percentage of nucleotides aligned to mitochondria from all the sequenced nucleotides. Also presented are the mean enrichment factors based on percentage and the mean genomic and mitochondria read lengths, for each triplicate of experiments, for each of three sequencing times, with and without the use of Read Until (selective sequencing)

Sequencing Time	Read Until	Mitochondria coverage	% Mitochondria nucleotides	Enrichment factor	Genomic read lengths	Mitochondria read lengths
10 min	No	3.26	0.088	2.30	6165	5965
	Yes	4.75	0.170		3450	4563
30 min	No	12.95	0.104	1.38	7598	8088
	Yes	13.78	0.143		5106	6382
120 min	No	37.60	0.112	3.12	8089	7570
	Yes	53.98	0.269		2488	4319

Table 3. Comparison of experimental and theoretically possible enrichment values of previously published adaptive sequencing methods for nanopore sequencing

Method	Experimental enrichment	Theoretical enrichment	Enrichment ratio
Loose et al. [24]	2.2 ^a	8.72	0.25
Edwards et al. [25]	1.14 ^b	8.84	0.13
Payne et al. [26]	2.84 ^a	5.54	0.51
Maio et al. [27]	5 ^a	7.1	0.70
Kovaka et al. [28]	4.46 ^b	11.64	0.38
Ours	2.3	2.9	0.79

^aCalculated enrichment value based on the results. ^bExperimental/reported enrichment value.

gradually over the years. Theoretical enrichment values vary widely between experiments, as the particular characteristics of each experiment affected the calculations dramatically. Finally, the experimental to theoretical ratio value constructs the best metric or real-world performance and also increased over the years; the use of our method gained greater value than the other methods.

Discussion

The process and the results of the deep learning model training

The overall high accuracy (>70%) of most of the models of deep learning training suggests that deep learning in general might be an appropriate solution for read classification based on raw signals. The higher accuracy of larger models is an expected outcome because larger networks have more weights that can be adjusted during the training process, and can possibly capture more variability of the data [49].

Interestingly, the smaller regular CNN model had higher overall accuracy than the medium and large CNN models. This can be explained by comparing all the CNN network configurations; the larger networks performed well (>90% accuracy) on some datasets, but on other datasets they either over-fit or did not train at all. However, the smaller CNN network demonstrated much lower accuracy but similar performance across all datasets, yielding a higher total accuracy. This is possibly due to the relative simplicity of a CNN model and the

fewer weights that smaller CNN models have to train. Thus, smaller CNN models need to generalize better than larger models, in agreement with findings in other studies [50].

VDCNN expectedly outperformed a regular CNN, as was shown in the original paper by Conneau *et al.* [37]. Another expected result is the RNN-type architectures (regular LSTM, LSTM with recurrent batch normalization and GRU), which outperformed the CNN-type architectures. The data in our study could be described as a sequential input, which is the type of data that RNN architecture was designed to analyze [6]. However, we observed similarity in the average accuracies of regular LSTM and VDCNN. This can be explained by the relative simplicity of the one-layered LSTM model compared to the more complex VDCNN with 17 layers. We expected the combination of the CNN + RNN model to outperform each type individually. This is because convolutional networks are useful for feature extraction [51], and the conjunction with RNN could be expected to improve results [52]. In our work, the combination of CNN + RNN produced results with similar accuracy to those of LSTM with recurrent batch normalization. These results can be explained by either the very optimal training of LSTM, with a recurrent batch normalization model; or by the sub-optimal training of the CNN + RNN models.

In choosing an appropriate model, training time and inference time should be important considerations, as they can greatly affect performance in a real-time application such as real-time selective sequencing. The training time of the models in our study differed, according to the model type and model size; the range was several minutes for the simple models to several hours for more complex models. The selected LSTM, with the recurrent batch normalization model, required 2 h for initial training (300 epoch) and an additional 10 min for each fine tuning (30 epoch). Inference time also changed, depending on the model, yet was in the range of a few milliseconds for all models. This is much less than the minimum response time of 1 s according to ONT [30]. Perhaps, for ultra-large models that have recently become prevalent and that require prolonged training on industrial level GPU servers [53], costs could be considered in selecting a model. However, as our work was performed on a consumer grade GPU, within a reasonable time range, training time and inference time were not considered in selection of the model.

Finetuning the models on a small portion of the dataset before analyzing the rest of the dataset improved results for some models, as seen in Table 1. Each dataset was acquired from a different sequencing experiment. Various variables could possibly affect the raw signal, such as differences in chemistry kits, MinION devices, library preparation protocols and sample qualities. Therefore, by finetuning the model to each experiment, we increased the model's accuracy for those specific conditions.

Dropout and batch normalization improved the performance of most models, as expected, based on their contribution to the training process of the deep learning models [6, 54]. In addition, before the addition of the difference transformation to the raw input, the results after training models were poor. Specifically, overfitting was a substantial problem before adding artificial noise, which is known to help with the training of deep learning models [55]. Therefore, those two transformations were applied to the training of all the models.

The mechanism by which the deep learning models perform classification remains unknown. The models could 'simply remember' the relatively short sequence of the mitochondria (only 16.5 K nucleotides) and determine the reads that originate from this sequence. Alternatively, the models could extract specific features from the reads such as GC content/k-mer content and more complex features such as the protein sequence and structure, or DNA methylations. Then, during training, the models could learn the features that are present in genomic sequences and those present in mitochondrial sequences. We also postulated that the models could have learned more sophisticated features of the mitochondrial DNA, such as different encoding codons, or the density of the genetic information [56]. Furthermore, deep learning models have been shown to successfully detect circular plasmids, based on their sequencing data, by examining larger chunks of the plasmid sequences achieved by longer reads, as well as additional genomic features [57]. This information could contribute to successful classification. We think that a thorough analysis of a trained deep learning model from this work, as was done for the visual analysis models [51], could provide useful insights for further research in this field. Moreover, new biological features that were not previously considered important could be discovered.

For 13 samples from various sources, the deep learning model correctly classified mitochondrial reads. This is despite the large variety in the experimental methods that produced the samples, such as DNA extraction and processing kits and methods, sequencing library kits and methods, flow-cell versions and sequencing device models. All these could have dramatically impacted the electrical signal that was produced during the sequencing. The accuracy of the model was relatively high among all human samples, even without performing finetuning for each dataset. Even though the different experimental conditions could affect the raw signal and impact the performance of classification, the successful classification of human samples from different sources demonstrate that the benefits of using a deep learning model outweigh its limitations and prove it is a viable method to use in future experiments. The discrepancy in accuracy for the various cell types is likely due to the large variation in the experimental parameters, though it could also be due to unknown differences between the cell types. However, validating the latter possibility would require more

sequencing data than is presently publicly available. The relatively high classification accuracy for samples from non-human organisms suggests that the deep learning model was able to learn ‘general’ patterns present in the mitochondria, rather than simply memorizing the human mitochondria sequence. The inverse correlation of accuracy with evolutionary distance further supports this idea. Specifically, mitochondrial sequences of organisms more distant on the evolutionary tree could differ more substantially from human sequences; and such patterns may not be recognized by the deep model.

Real-time selective sequencing

Our experiments utilized the ability of MinION to perform selective sequencing combined with a deep learning model, and demonstrated the validity of such method from several aspects. From the aspect of classification accuracy, our deep learning achieved >90% accuracy in real time. This is similar to results obtained from training and testing on the previous data. Due to the relatively small amount of mitochondrial DNA present in the sample, some variance was evident in the mitochondrial proportions between the experiments. Nonetheless, the significant differences between experiments with and without selective sequencing indicate that our method worked successfully, and correlated to the theoretically calculated expected enrichment results, as described below. Also, the differences between the experiments in the lengths of the mitochondrial and genomic reads support the notion that executing the selective sequencing script prioritized the mitochondrial reads over the genomic reads during sequencing.

To assess the results of our selective sequencing script, we can calculate the expected results in a theoretically perfect hardware–software configuration (in which each read that was marked for rejection would not have been sequenced), with a model of 90% accuracy ($\text{Acc} = 0.9$) for samples in which 0.1% of the reads are mitochondrial (similar to our samples $M_{\text{samp}} = 0.001$). We can calculate this theoretical expected mitochondrial percentage using the following formula (the expected ‘true’ mitochondrial reads divided by genomic reads are falsely classified as mitochondrial reads):

$$M_{\text{exp}} = \frac{(M_{\text{samp}} \times \text{Acc})}{(1 - M_{\text{samp}}) \times (1 - \text{Acc})}.$$

From this calculation, we can infer that selective sequencing in theoretically perfect conditions should yield 0.9% mitochondrial reads. Accordingly, we would achieve an enrichment of 9X when using selective sequencing and with perfect software-hardware performance.

To achieve an improved theoretical calculation, we utilized the official information provided by ONT [30]. The calculation takes into account three additional parameters: read lengths—lengths of the DNA molecules

in the sample; capture time—the time required to capture a new DNA molecule and sequencing speed—the speed of nucleotides passing through the pore. All three parameters could affect the final enrichment factor that was achieved in our experiments. ONT also provides an example of experimentally validated results, in which they calculated and achieved the same 5.2X enrichment factor. To calculate the expected enrichment factor in our experiment, we used 1.5 s, which is a slightly larger capture time than that used in the information sheet. This is because our library and flow cell are older and perhaps less efficient. In addition, we used a longer response time of 1.8 s as our deep learning implementation was not as well tuned as the official ONT implementation of adaptive sampling. However, we used the same sequencing speed of 420 nucleotides per second; and used the median of the mitochondrial and genomic read lengths from the HEK 1 experiment, which are similar to the median lengths used in the information sheet. Taking into account all those parameters and the accuracy of the model, which was 90%, we calculated an expected enrichment factor of X2.9. In our experiments, we achieved enrichment of 2.3X, which is very close to the calculated result. The lower enrichment factor could possibly be explained by deterioration during the experiment, in a similar fashion as in an ONT run. Interestingly, increasing the ‘number of batched reads’ from 10 to 100 yielded an increase in enrichment factor (see the ‘Additional selective sequencing experiment’ section of the Supplementary Material, see Supplementary Data available online at <http://bib.oxfordjournals.org/>). This change might drastically affect deep learning performance as deep learning models are optimized to operate with larger batches of reads. Repeating the calculations after decreasing the response time by 10, to 0.2 s, yielded an enrichment factor of 3.82X, as expected. This compares with the experimentally obtained enrichment of 3.8X, thus concurring with our theoretical explanation of the higher response time in the experiments. The preferred response time of 0.2 s, combined with the highest accuracy of a deep learning model, achieved accuracy of 98.6% for our experiments. With the capture time of current reagents and flow cells of 1.2, we can calculate a possible enrichment factor of 8.6X. This theoretical calculation enables directly comparing our enrichment results to the traditional adaptive sequencing methods, as those methods are based on the alignment of the genome to a reference. Utilizing the first 200 nucleotides of the reads, the accuracy of such algorithms would be higher than 99%.

The percentages of mitochondrial nucleotides sequenced with and without selective sequencing differed significantly. The smaller difference between the percentages of mitochondrial reads in experiments with and without selective sequencing, compared to the differences in the genomic reads with and without selective sequencing, also supports the notion that

selective sequencing prioritizes the mitochondrial reads. Our claim that our selective sequencing method yielded more mitochondrial sequences is further supported by the result obtained when we combined the differences in the raw mitochondrial nucleotide counts without normalization. This showed the sequencing of more mitochondrial sequences in experiments with selective sequences. Therefore, even in its current state, our approach could assist researchers in achieving better coverage of a certain region, and theoretically save time and resources by requiring less sequencing to achieve a similar goal. We conclude that the genomic reads were shorter in experiments with selective sequencing, probably because our script sent signals to the MinION device to stop sequencing the reads that were classified as genomic. Thus, non-mitochondrial reads would be shorter than in experiments without the stopping signal.

Comparing our method to other adaptive sequencing methods provides further insights. The experimental enrichment values provide a 'dry' metric of the method's performance, while the experimental to theoretical enrichment ratio gives a more useful metric, describing performance in the context of experimental parameters. In our experiments, the extremely small portion of the targeted sequences (mitochondrial sequences) in the samples, together with the lower accuracy of classification compared to other methods, produced a relatively low expected enrichment value. Accordingly, the experimental to theoretical enrichment ratio of our work was higher than that of other methods. The higher experimental to theoretical enrichment ratio suggests that our method could perform better adaptive sequencing when run with similar experimental characteristics as other experiments.

Conclusion

From the results of the deep learning models training, we conclude that this approach is a valid choice for classifying sequenced reads based on the first 2000 values of raw signal of the read. There might be better models than those we tested here; however, even our relatively simple and straightforward approach demonstrated good results in terms of accuracy and generalization, for a number of datasets. In addition, testing 13 samples from various origins provided additional support of the generalization of our deep learning model to various cell types, as well as to other closely related organisms. Furthermore, for the first time, we showed the ability of deep learning models to classify whole reads based on raw nanopore signals.

The selective sequencing experiments performed with our script, using the best deep learning model from the previous steps, produced sufficient evidence to conclude that our script prioritized mitochondrial reads over genomic reads. The deep learning model classified the reads correctly while they were being sequenced; analysis of the proportion of mitochondrial DNA and the

differences in read lengths revealed that the mitochondrial reads were prioritized during sequencing and were enriched by a factor of 2.3X.

Taken together, the results from both parts of the experiment demonstrated the feasibility of real-time selective sequencing, using deep learning models that analyze the raw signal at the beginning of each read. Comparison to other adaptive sequencing methods also supports the notion that our method could perform comparatively or better compared to other methods even when using the same experimental characteristics. Better models to perform classification with increased accuracy may be developed, since our models were basic versions of each architecture and many improvements in deep learning methods have been introduced.

We believe that our findings provide a solid basis for future research in this field. We hope that our deep learning models will serve as a building block for studies on improving the analysis of raw signals. The technique may be used as an alternative to other adaptive sequencing methods particularly in tasks in which traditional methods using a reference are not feasible. This is because the difference between the DNA molecules is in methylation or in other parameters, such as nucleotide composition, and the nucleotides and the ends of the DNA molecule that could differ in pathological or cancer DNA [58–61]. Our method could spark interest in improving the selective sequencing optimization to provide a much better procedure. Our study might also serve as an alternative to other selective sequencing methods.

Key Points

- The novel use of deep learning in classifying raw nanopore sequencing data of entire DNA molecules.
- A 90–98% accuracy for classifying mitochondrial DNA versus genomic DNA.
- Successful classification of DNA from different datasets and organisms.
- Successful use of the nanopore Read Until/Adaptive Sampling feature.
- Experimentally confirmed enrichment by 2.3-fold of mitochondrial DNA, as predicted by theoretical calculations.
- Code available at https://github.com/nshomron/Nanopore_Deep_Learning.

Supplementary data

Supplementary data are available online at <http://bib.oxfordjournals.org/>.

Authors' Contribution Statement

A.D. conceived, designed and led the project; he implemented the computational part of the project and wrote the manuscript. A.P. developed an optimized sequencing library workflow, implemented the experimental part of

the project and wrote the manuscript. N.S. conceived, designed and led the project, and wrote the manuscript.

Acknowledgments

We thank Dr David Golan, Prof Lior Wolf and Tzviel Frostig from Prof Yoav Benjamini's laboratory for consultations. Figure 1 was created with BioRender.com.

Funding

Edmond J. Safra Center for Bioinformatics at Tel-Aviv University, Gertner Institute, Zimin Institute, Tel Aviv University Innovation Laboratories (TILabs), and the Djerassi-Elias Institute of Oncology.

Data Availability

The sequencing data produced in this study are available at NCBI's SRA with accession number PRJNA689046.

Code Availability

The code used for this study is provided for non-commercial use at: https://github.com/nshomron/Nanopore_Deep_Learning.

References

- Goodwin S, McPherson JD, McCombie WR. Coming of age: ten years of next-generation sequencing technologies. *Nat Rev Genet* 2016;**17**:333–51.
- Jain M, Olsen HE, Paten B, et al. The Oxford Nanopore MinION: delivery of nanopore sequencing to the genomics community. *Genome Biol* 2016;**17**:239.
- Rang FJ, Kloosterman WP, de Ridder J. From squiggle to basepair: computational approaches for improving nanopore sequencing read accuracy. *Genome Biol* 2018;**19**:90.
- Rand AC, Jain M, Eizenga JM, et al. Mapping DNA methylation with high-throughput nanopore sequencing. *Nat Methods* 2017;**14**:411–3.
- Simpson JT, Workman RE, Zuzarte PC, et al. Detecting DNA cytosine methylation using nanopore sequencing. *Nat Methods* 2017;**14**:407–10.
- LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature* 2015;**521**:436–44.
- Angermueller C, Pärnamaa T, Parts L, et al. Deep learning for computational biology. *Mol Syst Biol* 2016;**12**:878.
- Ching T, Himmelstein DS, Beaulieu-Jones BK, et al. Opportunities and obstacles for deep learning in biology and medicine. *J R Soc Interface* 2018;**15**:20170387.
- Çakır E, Parascandolo G, Heittola T, et al. Convolutional recurrent neural networks for polyphonic sound event detection. *IEEE/ACM Trans Audio Speech Lang Process* 2017;**25**:1291–303.
- Lee H, Pham P, Largman Y, et al. Unsupervised feature learning for audio classification using convolutional deep belief networks. In: Bengio Y, Schuurmans D, Lafferty JD, et al. (eds). *Advances in Neural Information Processing Systems*, Vol. 22. Curran Associates, Inc, 2009, 1096–104.
- Huang P, Kim M, Hasegawa-Johnson M, et al. Deep learning for monaural speech separation. In: 2014 *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 1562–61566. <https://doi.org/10.1109/ICASSP.2014.6853860>
- David M, Dursi LJ, Yao D, et al. Nanocall: an open source basecaller for Oxford Nanopore sequencing data. *Bioinformatics* 2017;**33**:49–55.
- Timp W, Comer J, Aksimentiev A. DNA Base-Calling from a Nanopore Using a Viterbi Algorithm. *Biophys J* 2012;**102**:L37–9.
- Teng H, Cao MD, Hall MB, et al. Chiron: translating nanopore raw signal directly into nucleotide sequence using deep learning. *GigaScience* 2018;**7**:giy037.
- Boža V, Brejová B, Vinař T. DeepNano: Deep recurrent neural networks for base calling in MinION nanopore reads. *PLoS One* 2017;**12**:e0178751.
- Ni P, Huang N, Zhang Z, et al. DeepSignal: detecting DNA methylation state from Nanopore sequencing reads using deep-learning. *Bioinformatics* 2019;**35**:4586–95.
- Li Y, Han R, Bi C, et al. DeepSimulator: a deep simulator for Nanopore sequencing. *Bioinformatics* 2018;**34**:2899–908.
- Ng SB, Turner EH, Robertson PD, et al. Targeted capture and massively parallel sequencing of 12 human exomes. *Nature* 2009;**461**:272–6.
- Gnirke A, Melnikov A, Maguire J, et al. Solution hybrid selection with ultra-long oligonucleotides for massively parallel targeted sequencing. *Nat Biotechnol* 2009;**27**:182–9.
- Tewhey R, Warner JB, Nakano M, et al. Microdroplet-based PCR enrichment for large-scale targeted sequencing. *Nat Biotechnol* 2009;**27**:1025–31.
- Karamitros T, Magiorkinis G. Multiplexed targeted sequencing for oxford nanopore MinION: a detailed library preparation procedure. In: Head SR, Ordoukhanian P, Salomon DR (eds). *Next Generation Sequencing: Methods and Protocols*. New York, NY: Springer, 2018, 43–51. <https://doi.org/10.1007/978-1-4939-7514-3-4>
- Gabrieli T, Sharim H, Fridman D, et al. Selective nanopore sequencing of human BRCA1 by Cas9-assisted targeting of chromosome segments (CATCH). *Nucleic Acids Res* 2018;**46**:e87.
- Mertes F, ElSharawy A, Sauer S, et al. Targeted enrichment of genomic DNA regions for next-generation sequencing. *Brief Funct Genomics* 2011;**10**:374–86.
- Loose M, Malla S, Stout M. Real-time selective sequencing using nanopore technology. *Nat Methods* 2016;**13**:751–4.
- Edwards HS, Krishnakumar R, Sinha A, et al. Real-time selective sequencing with RUBRIC: Read Until with basecall and reference-informed criteria. *Sci Rep* 2019;**9**:1–11.
- Payne A, Holmes N, Clarke T, et al. Readfish enables targeted nanopore sequencing of gigabase-sized genomes. *Nature biotechnology* 2021;**39**:442–450.
- Maio ND, Manser C, Munro R, et al. BOSS-RUNS: a flexible and practical dynamic read sampling framework for nanopore sequencing. *bioRxiv*. 2020. <https://doi.org/10.1101/2020.02.07.938670>.
- Kovaka S, Fan Y, Ni B, et al. Targeted nanopore sequencing by real-time mapping of raw electrical signal with UNCALLED. *Nat Biotechnol* 2021;**39**:431–41.
- Chen R, Aldred MA, Xu W, et al. Comparison of whole genome sequencing and targeted sequencing for mitochondrial DNA. *Mitochondrion* 2021;**58**:303–10. <https://doi.org/10.1016/j.mito.2021.01.006>
- Community - Info sheet - adaptive-sampling (Resource available after a free registration). https://community.nanoporetech.com/info_sheets/adaptive-sampling/v/ads_s1016_v1_rev12

- nov2020/considerations-for-experimental-design (16, March, 2022, date last accessed).
31. Jain M, Koren S, Miga KH, et al. Nanopore sequencing and assembly of a human genome with ultra-long reads. *Nat Biotechnol* 2018;**36**:338–45.
 32. Consortium, T. 1000 G. P. A global reference for human genetic variation. *Nature* 2015;**526**:68–74.
 33. Brown C. ONT-HG1. 2017. <https://doi.org/10.5281/zenodo.1318628>
 34. Masters D, Luschi C. Revisiting small batch training for deep neural network. *arXiv preprint arXiv:1804.07612*. 2018.
 35. Paszke A, Gross S, Massa F, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In: *Advances in Neural Information Processing Systems 32 [Internet]*. Curran Associates, Inc.; 2019. p. 8024–35.
 36. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. In: Pereira F, Burges CJC, Bottou L, Weinberger KQ (eds). *Advances in Neural Information Processing Systems*, Vol. 25. Curran Associates, Inc., 2012, 1097–105.
 37. Conneau A, Schwenk H, Barrault L, et al. Very deep convolutional networks for text classification. *arXiv preprint arXiv:1606.01781* 2016.
 38. Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput* 1997;**9**:1735–80.
 39. Cooijmans T, Ballas N, Laurent C, et al. Recurrent batch normalization. *arXiv preprint arXiv:1603.09025* 2016.
 40. Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint. arXiv:1406.1078* 2014.
 41. Sainath TN, Vinyals O, Senior A, et al. Convolutional, long short-term memory, fully connected deep neural networks. In: 2015 *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015. pp.4580–4. <https://doi.org/10.1109/ICASSP.2015.7178838>
 42. Kingma DP, Ba JA. A method for stochastic optimization. *ArXiv14126980 Cs* 2014.
 43. Yosinski J, Clune J, Bengio Y, et al. How transferable are features in deep neural networks? In: Ghahramani Z, Welling M, Cortes C et al. (eds). *Advances in Neural Information Processing Systems*, Vol. 27. Curran Associates, Inc, 2014, 3320–8.
 44. Kumar S, Stecher G, Suleski M, et al. TimeTree: a resource for timelines, timetrees, and divergence times. *Mol Biol Evol* 2017;**34**: 1812–9.
 45. Wick RR, Judd LM, Holt KE. Performance of neural network basecalling tools for Oxford Nanopore sequencing. *Genome Biol* 2019;**20**:129.
 46. Li H. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics*. 2018;**34**:3094–3100.
 47. Jaeger TF. Categorical data analysis: away from ANOVAs (transformation or not) and towards logit mixed models. *J Mem Lang* 2008;**59**:434–46.
 48. Boik RJ. The Fisher-Pitman permutation test: a non-robust alternative to the normal theory F test when variances are heterogeneous. *Br J Math Stat Psychol* 1987;**40**:26–42.
 49. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* 2014.
 50. Zhang C, Bengio S, Hardt M, et al. Understanding deep learning requires rethinking generalization. *Communications of the ACM* 2021;**64**:107–115.
 51. Zeiler MD, Fergus R. Visualizing and understanding convolutional networks. In *European conference on computer vision*. Springer, Cham. 2014 (pp. 818–833).
 52. Ordóñez F, Roggen D, Ordóñez FJ, et al. Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. *Sensors* 2016;**16**:115.
 53. Brown T, Mann B, Ryder N, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 2020;**33**:1877–1901.
 54. Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning* 2015. (pp. 448–456). PMLR.
 55. Neelakantan A, Vilnis L, Le QV, et al. Adding gradient noise improves learning for very deep networks. *arXiv preprint arXiv:1511.06807*. 2015.
 56. Taanman JW. The mitochondrial genome: structure, transcription, translation and replication. *Biochim Biophys Acta BBA Bioenerg* 1999;**1410**:103–23.
 57. Andreopoulos WB, Geller AM, Lucke M, et al. Deepplasmid: deep learning accurately separates plasmids from bacterial chromosomes. *BioRxiv* 2021.03.11.434936. 2021. <https://doi.org/10.1101/2021.03.11.434936>.
 58. Kim H, Jen J, Vogelstein B, et al. Clinical and pathological characteristics of sporadic colorectal carcinomas with DNA replication errors in microsatellite sequences. *Am J Pathol* 1994;**145**: 148–56.
 59. Simón D, Cristina J, Musto H. Nucleotide composition and codon usage across viruses and their respective hosts. *Front Microbiol* 2021;**12**:1742.
 60. Brennan EP, Ehrlich M, Brazil DP, et al. Comparative analysis of DNA methylation profiles in peripheral blood leukocytes versus lymphoblastoid cell lines. *Epigenetics* 2009;**4**: 159–64.
 61. Jiang P, Xie T, Ding SC, et al. Detection and characterization of jagged ends of double-stranded DNA in plasma. *Genome Res* 2020;**30**:1144–53.