DeepSimulator: a deep Nanopore sequencing simulator

Yu Li[1], Renmin Han[1], Chongwei Bi[2], Mo Li[2], Sheng Wang[1*], Xin Gao[1*]
[1]Computational Bioscience Research Center (CBRC),
[2]Biological and Environmental Sciences and Engineering (BESE) Division,
King Abdullah University of Science and Technology (KAUST), Thuwal 23955-6900, Saudi Arabia
* realbigws@gmail.com or xin.gao@kaust.edu.sa

Abstract

Motivation: Oxford Nanopore sequencing is a rapidly developed sequencing technology in recent years. To keep pace with the explosion of the downstream data analytical tools, a versatile Nanopore sequencing simulator is needed to complement the experimental data as well as to benchmark those newly developed tools. However, all the currently available simulators are based on simple statistics of the produced reads, which have difficulty in capturing the complex nature of the Nanopore sequencing procedure, the main task of which is the generation of raw electrical current signals.

Results: Here we propose a deep learning based simulator, DeepSimulator, to mimic the entire pipeline of Nanopore sequencing. Starting from a given reference genome or assembled contigs, we simulate the electrical current signals by a context-dependent deep learning model, followed by a base-calling procedure to yield simulated reads. This workflow mimics the sequencing procedure more naturally. The thorough experiments performed across four species show that the signals generated by our context-dependent model are more similar to the experimentally obtained signals than the ones generated by the official context-independent pore model. In terms of the simulated reads, we provide a parameter interface to users so that they can obtain the reads with different accuracies ranging from 83\% to 97\%. The reads generated by the default parameter have almost the same properties as the real data. Two case studies demonstrate the application of DeepSimulator to benefit the development of tools in *de novo* assembly and in low coverage SNP detection.

Availability: The software can be accessed freely at: https://github.com/lykaust15/DeepSimulator.

1 Workflow of DeepSimulator

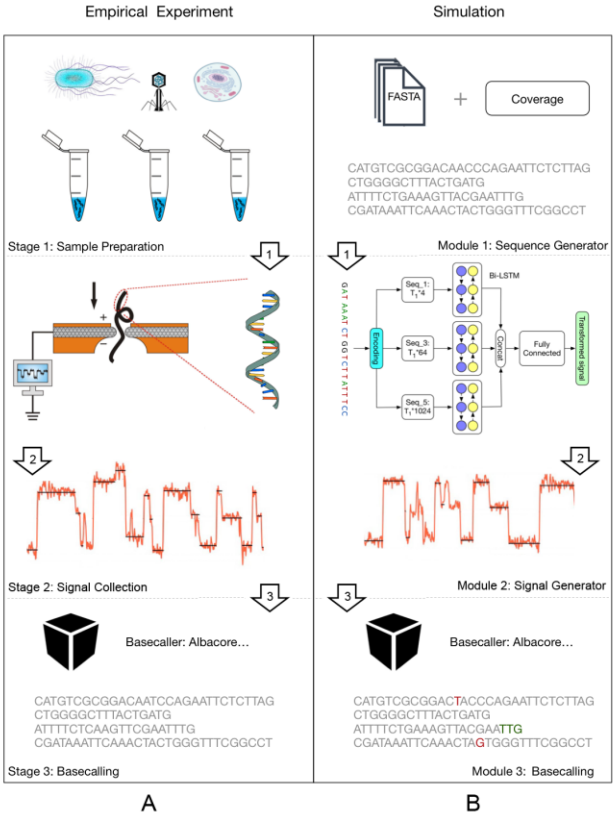

Figure 1. (A) The Nanopore sequencing procedure. (B) The main workflow of DeepSimulator. It simulates the entire pipeline of the empirical Nanopore sequencing experiment, producing both the simulated signals and the final simulated reads. In addition, DeepSimulator is highly modularized, which means it can be customized and updated easily to keep up with the development pace of the Nanopore sequencing technologies. Unlike the real data, the ground truth and the annotation of the simulated reads are easy to acquire. In the simulated reads on the bottom of the figure, the red colored bases are the mismatches. The green colored bases indicate that there are indel (insertion and deletion) before them.

2 Key algorithm

It is obvious that the core component of our simulator is the pore model in the signal generation module. Currently, all the existing pore models are context-independent, which assign each 5-mer a fixed value for the expected current signal regardless of its location on the nucleotide sequence.

In order to further polish our simulator, we propose a novel context-dependent pore model, taking advantage of deep learning techniques, which have shown great potential in bioinformatics. Nonetheless, it is not straightforward to train the deep learning model because of the fact that the current signal is usually 8~10 times longer than the nucleotide sequence.

To conquer this difficulty, we propose a novel deep learning strategy, BiLSTM-extended Deep Canonical Time Warping (BDCTW), which combines bi-directional long short-term memory (Bi-LSTM) [Graves *et al.*, 2005] with deep canonical time warping (DCTW) [Trigeorgis *et al.*, 2016] to solve the scale difference issue.

2.1 General framework of deep canonical time warping

The goal of deep canonical time warping (DCTW) is to discover a hierarchical or recurrent non-linear relationship between two input linearly structured datasets X_1 and X_2 with different lengths T_1, T_2 and feature dimensionality d_1, d_2 (i.e. X_i). That is, DCTW simultaneously performs spatial transformation and temporal alignment between the two input data sequences.

In our case, the two inputs are the nucleotide sequence X and the observed electrical current signal sequence Y. As shown in Figure 2, after DCTW, the transformed features from X and Y are not only temporally aligned with each other, but also maximally correlated. To this end, let us consider that $Y_i = F_i(X_i\,;\,\theta_i)$ representing the activation function of the final layer of the corresponding deep neural network (DNN, such as Bi-LSTM) for X_i, which has d maximally correlated units where $d \leq \min(d_1,d_2)$. Such an operation reduces the input data samples to the same feature dimension and then performs a maximal correlation analysis, which essentially resembles the classical canonical correlation analysis (CCA) [Akaike, 1976].

$$\operatorname{argmin}_{\theta_1,\theta_2,\Delta_1,\Delta_2} \|F_1(X_1;\theta_1)\Delta_1 - F_2(X_2;\theta_2)\Delta_2\|_F^2$$

$$\begin{gathered}\text{Subject to: } F_i(X_i;\theta_i)\Delta_i \mathbf{1}_T = \mathbf{0}_d \\ F_i(X_i;\theta_i)\Delta_i\Delta_i^{\mathsf{T}}F_i(X_i;\theta_i)^{\mathsf{T}} = \mathbf{I}_d \\ F_1(X_1;\theta_1)\Delta_1\Delta_2^{\mathsf{T}}F_2(X_2;\theta_2)^{\mathsf{T}} = \mathbf{D}_d \\ \Delta_i \in \{0,1\}^{T_i \times T}, i = \{1,2\}\end{gathered}$$

where X_1=X, X_2=Y. T_1, T_2 and T are the length of X,Y and the final alignment, respectively. Δ_i are the binary selection matrices that encode the alignment paths for X_i. That is, D_1 and D_2 remap the nucleotide sequence X with length T_1 and raw signals Y with length T_2 to a common temporal scale T. T_2 to a common temporal scale T. **D** is a diagonal matrix. **I** is the identity matrix. And **1** (**0**) is an appropriate dimensionality vector of all 1's (0's).

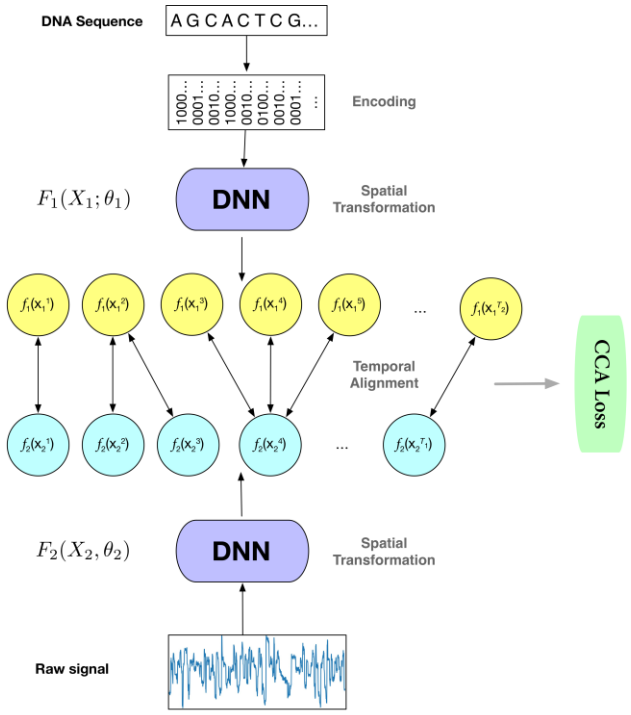

Figure 2. Illustration of the deep canonical time warping (DCTW) architecture with two deep neural networks (DNNs), one for the input nucleotide sequence (here we use one-hot encoding for each nucleotide and thus the feature dimension is four) and the other for the observed electrical current measurements (denoted as raw signals with feature dimension one). We train this model in an end-to-end manner, which first performs a spatial transformation that efficiently reduces the input data samples to the same feature dimension, followed by a temporal alignment that effectively maps the samples of each input sequence to a common temporal scale. The objective function of the model is to make the transformed input data samples to be maximally correlated under the canonical correlation analysis (CCA) loss.

References

[Trigeorgis *et al.*, 2016] George Trigeorgis, Mihalis A. Nicolaou, Bjorn W. Schuller, Stefanos Zafeiriou. Deep canonical time warping. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5110–5118. 2016.

[Akaike, 1976] Akaike,H. Canonical correlation analysis of time series and the use of an information criterion. *Math. Sci. Eng.*, 126, 27–96. 1976.

[Graves *et al.*, 2005] Alex Gravesa and Jurgen Schmidhuber. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Netw.*, 18, 602-610.