REVIEW OF BASECALLING METHODS FOR THIRD GENERATION SEQUENCING DEVICES

Pavan Chennagiri (UT EID: pm29224)

Electrical and Computer Engineering,
The University of Texas at Austin, Austin, Texas, USA
pavancm@utexas.edu

ABSTRACT

Sequencing by DNA translocation is an emerging technology which offers cheaper and faster devices for DNA sequencing. However, precisely determining the bases present in the sequence from noisy and lengthy electric signals is particularly challenging and offers an interesting research problem. In this project two basecalling methods DeepNano and Chiron are evaluated and the fundamental advantages and drawbacks associated with them are analyzed. Chiron uses a combination of CNN-RNN model as opposed to RNN in case of Deepnano. Comparison between the methods is made in the context of performance accuracy,speed, complexity and generalizability. Chiron with it's complex model provides better read accuracy as well as generalizes well with unseen data when compared to Deepnano.

***Index Terms*—** DNA basecalling, third generation sequencing, nanopore technology, Recurrent Neural Networks, Convolutional Neural Networks, deep basecallers

1. INTRODUCTION

The MinION device by Oxford Nanopore Technologies (ONT) is the first portable sequencing device employing nanoscaled pores technology to sequence DNA. The MinION device weighing only 90 grams, is currently the smallest high-throughput DNA sequencer. With its low capital costs, small size and the possibility of analyzing the data in real time by sequencing tens of thousands of base pairs. The important innovation in ONT is it's ability to measure the changes in electric current across the pore when DNA passes through it. MinION uses nanoscaled pores to sequence DNA. An electrical potential is applied over a membrane in which a pore is inserted. As the DNA passes through the pore, the sensor detects changes in ionic current caused by different nucleotides present in the pore [1]. The electric current is measured at each pore several thousand times per second, resulting in a measurement plot as shown in Fig. 1.

A MinION device results in reads which typically contain several thousand base pairs. Although MinION is able to produce extremely long reads, it is challenging to decrypt the bases from these noisy and complex electric signals. In order to reduce the error rate, the device attempts to read both strands of the same DNA fragment. The resulting template and complement reads can be combined to a single two-directional (2D) read during base calling. In this project only 1D reads are analyzed (generally 2D reads are associated with better performance when compared to 1D counterparts [2], hence improving 1D performance leads corresponding improvement in 2D reads). When MinION was first introduced it's technology was labeled as R7 chemistry, subsequently it's technology was updated to R9 chemistry resulting in better performance. In this project all the data that is analyzed were obtained from R9.4 (updated version of R9) chemistry.

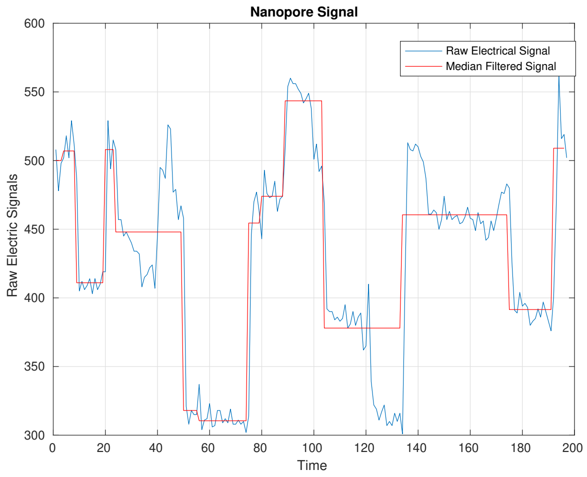

Fig. 1: Illustration of raw nanopore signal along with median filtered version. Different piecewise constant lengths correspond to different bases present in the sequence

This type of sequencing is sometimes referred to as third generation sequencing since its method differs from Sangers first generation and second generation shotgun sequencing. When MinION was first introduced by ONT, it's in-house basecaller (known as Metrichor) employed a HMM for base-

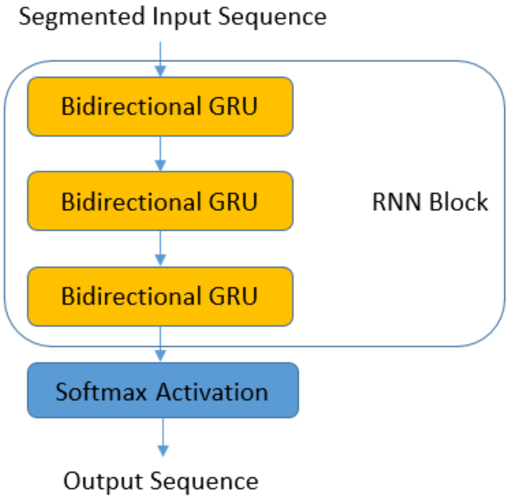

Fig. 2: DeepNano architecture

calling. It assumed that the electric signal can be segmented into 'events' with each event representing a k-mer, and consecutive event changes resulted in change in nucleotides (k-1 nucleotides overlap between consecutive k-mers). This algorithm is ONT's proprietary basecaller and it's exact details are unknown. Similar approach was used in Nanocall [3] as well. One of the major drawbacks of HMM is it's inability to capture long range dependencies which are significant in ultra long reads produced by nanopores.

DeepNano [2] was the first open source basecaller which proposed employing a Recurrent Neural Network (RNN) for basecalling. Motivation to employ RNN comes from it's successful results in speech recognition, language modeling and other sequence processing tasks. In DeepNano the electric signals are first segmented into events and certain simple features such as mean, variance are extracted from each event and then fed to RNN to obtain the nucleotide sequence. DeepNano employs a bidirectional RNN as the nucleotide dependency can extend both ways.

Segmenting the raw nanopore electrical signal into segments is in general non trivial and error-prone approach. Segmentation algorithms determine a boundary between two segments based on a sharp change of signal values, which can be erroneous given that raw electrical signals are particularly noisy. In Chiron basecaller [4] a Convolutional Neural Network (CNN) is proposed in place of segmentation followed by a bidirectional RNN. In other words Chiron takes raw electric signals as input as opposed to segmented events in order to determine base sequence.

In this project Chiron and DeepNano basecallers are studied in detail, and a comparison is made between them by performing various experiments across different criteria. The rest of the report is organized as follows. In Sec 2 methodology along with the architecture and databases employed for evaluation is discussed. Various experiments and results are reported in Sec 3. In Sec 4 conclusion is made followed by some thoughts on future work in Sec 5.

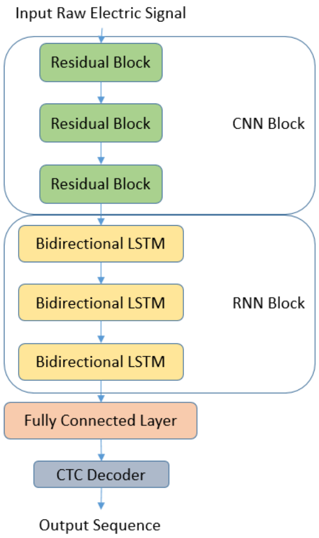

Fig. 3: Chiron Architecture with residual blocks. The contents present in the residual block is shown in Fig. 4

2. METHODS

In this section details of the models employed in Chiron and DeepNano is provided. A high level overview of the model architectures and the reasoning behind their use is also discussed. Lastly a brief description about the databases and the methods used to train and evaluate models is discussed.

2.1. Model Architecture

2.1.1. DeepNano

Fig. 2 shows the various blocks present in DeepNano. The model requires input sequence to be segmented into events as illustrated in Fig. 1. The segmented sequence is then fed to a RNN which consists of 3 layers of Bidirectional Gated Recurrent Units (GRU). RNNs typically model long range dependencies which are inherent to base sequence of the genomes. Bidirectional units are employed as the choice of the nucleotide is influenced by preceding as well as succeeding bases. Recently GRUs have been shown to perform well particularly in sequence modeling tasks [5]. The last step consists of softmax activation to obtain probability for each nucleotide and the base with highest probability is predicted as output. Cross entropy loss function is used to train the model for calculating weights of the hidden units.

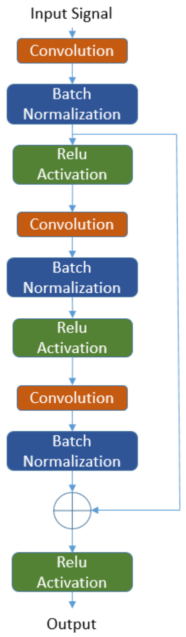

Fig. 4: Residual Block

2.1.2. Chiron

Fig. 3 illustrates the high level block diagram of Chiron basecaller. Chiron takes raw electric signals obtained from nanopores as input. This model is mainly motivated from [6], which had considerable success in speech recognition. The first step is local feature extraction which is accomplished by a CNN. In Chiron a modified version of CNN known as Residual Network [7] is used. Residual networks were primarily developed to address the problem of vanishing gradients in networks employing large number of hidden layers. Residual Block differs from CNN by having a skip connection as shown in Fig. 4. The locally extracted features are then fed to RNN. The RNN block is very similar to the one present in DeepNano except that Bidirectional Long Short Term Memory (LSTM) cells are used in place of GRU. Lastly a Connectionist Temporal Classification (CTC) [8] loss function is employed to train the model. During testing phase of the network, a decoder is needed for obtaining predicted bases from the CTC loss function.

2.1.3. Architecture Comparison

The fundamental difference between Chiron and Deepnano arise from the way they accept input. Deepnano requires event segmentation and certain handcrafted features are extracted from every segment. In contrast, Chiron takes raw electrical signals and predicts the nucleotide sequence without need of any segmentation step. Another major difference is the loss functions that ae used for training, Deepnano uses a simple cross-entropy loss while Chiron uses a CTC loss, the latter has been shown to have superior performance in sequence modeling tasks. Inside RNN, Deepnano uses a GRU cells, while LSTM is employed in case of Chiron. In the literature GRU and LSTM have been shown to have almost similar performance [5], although training a GRU is computationally less expensive when compared to that of an LSTM.

2.2. Dataset

I used a dataset containing reads of E.Coli and Phage Lambda released by the authors of Chiron [9], with each having 4000 reads. Each read on an average consists of 6000 base pairs. All the reads were obtained from ONT cells employing R9.4 chemistry. The reads were divided into multiple chunks of fixed size and each chunk was normalized by subtracting mean and diving by the standard deviation. Since training requires labeled data, labeling was done using Metrichor (ONT proprietary service) followed by Nanoraw [10] and the labeled data is made publicly available by the authors of Chiron.

2.3. Training

Training was done on a mixed dataset containing 2000 reads from each E.Coli and Phage Lambda (hence a total of 4000 reads were used for training). Since labeled data is needed for training, in case of Chiron only the segments of raw signal corresponding to nucleotides were used. In case of Deepnano, for each segment mean, standard deviation and length of the segment were extracted for training. Both the models were trained using a Nvidia Tesla P100 GPU, with Chiron requiring approximately 6 hours to complete 3 epochs while DeepNano taking less than 30 minutes for the same. Although Chiron converged in 3 epochs, training was continued until 6 epochs in case of Deepnano [1] as it took more epochs to converge. Higher training time for Chiron is attributed to the presence of larger number of parameters when compared to that of Deepnano. In both cases Stochastic Gradient Descent (SGD) optimization applied on minibatches of data was used.

2.4. Model Evaluation

The remaining 2000 reads of E.Coli and Phage Lambda were used for evaluating the performance of the trained models. In order to have a better accuracy each read was divided into blocks of 300 time units with an overlap of 240 units (80%) between two consecutive blocks. In overlapping regions consensus of predicted bases were used to obtain the read sequence. To assess the performance of the basecalled reads, the resulting FASTA/FASTQ reads were aligned to the reference genome using graphmap algorithm [11] with default parameters. The resulting BAM file was then assessed using the

[1] The performance reported in this report for Deepnano might not completely match with the results in their publication as the training datasets employed are different.

japsa error analysis tool (jsa.hts.errorAnalysis), which looks at the insertion,mismatch and deletion rates; and the identity rate compared to the reference genome. The identity rate was calculated as $\frac{\text{number of matched bases}}{\text{number of bases in reference}}$ and is the metric used here for evaluating basecalling read accuracy.

3. EXPERIMENTS AND RESULTS

This section contains the details of the experiments I conducted to evaluate the basecallers and the results of these experiments have been reported. A comparison is made between the results of the two basecallers and a possible reasoning behind their performance difference is also presented.

3.1. Read Accuracy

	Chiron	Deepnano
E.Coli	**84.22**	67.49
Lambda	**82.33**	70.53

Table 1: Identity rate for basecallers across different datasets. All values represent percentage identity rate.

Table 1 depicts the identity rate for each basecaller across the databases. Identity rate is calculated as $\frac{\text{number of matched bases}}{\text{number of bases in reference}}$. It can be clearly inferred that Chiron has superior accuracy when compared to Deepnano across all databases. This is expected as Chiron has a more complex model involving parameters for both segmentation stage (CNN block) as well labeling stage (RNN block) and employs no additional features apart from raw electric signal as input. On the other hand Deepnano is a simplistic model with lesser number of parameters, using extracted features from the raw electric signal as input. In other words Chiron efficiently *learns features* for segmentation in an automatic fashion which gives best results for labeling task when compared to handcrafted features used in Deepnano. The results reported are for default hyperparameter values recommended by the authors [2].

3.2. Basecalling Speed

Time taken to obtain inference from the trained network in testing phase for basecallers is known as basecalling speed. Table 2 summarizes the time taken by different basecallers for obtaining base sequence for 2000 reads from each dataset. As Deepnano network is simplistic in nature and involves no separate decoder after the output layer, running it on a GPU with parallel threads achieves speeds almost 10 times that of Chiron. On the other hand, Chiron is considerably slower due to it's complex network as well as employing a beam search decoding stage after the output layer resulting in more processing time.

	Chiron (s)	Deepnano (s)
E.Coli	25,125	**2,067**
Lambda	16,794	**1,852**

Table 2: Time taken for basecalling 2000 reads of each dataset in seconds. There is difference between time taken for different dataset as the number of bases present in 2000 reads across different datasets.

3.3. Generalization

The two models discussed in this project are essentially data driven approaches. So the question arises on how well these models perform on unseen data. This is particularly essential when sequencing reads which currently have no reference and hence cannot be used while training. Also it is impractical to include all reads from the existing sequenced genomes while training. In order to evaluate the generalization capability the basecallers were trained on one dataset and tested on another dataset. The results are shown in Table 3 and 4. Chiron remarkably has similar performance to that shown in Table 1 indicating a greater generalization capability. However Deepnano has a relatively poor performance indicating the model has considerable dependence on training data.

	Chiron	Deepnano
E.Coli	**84.25**	32.368

Table 3: Identity rates when trained on Lambda dataset and tested on E.Coli dataset

	Chiron	Deepnano
Lambda	**81.42**	29.809

Table 4: Identity rates when trained on E.Coli dataset and tested on Lambda dataset

4. DISCUSSION AND CONCLUSION

MinION device with it's portable nanopore technology has drastically reduced the cost of sequencing high-throughput DNA. However obtaining base sequence from nanopore reads is a challenging task and is currently less accurate. In this project a detailed analysis and comparison was made between two basecallers: Chiron and deepnano, which primarily address the problem of basecalling for nanopore reads. Chiron uses a combination of CNN-RNN model with raw electric signals as input for sequencing bases, while deepnano uses only RNN with features such as mean, standard deviation and length from each segmented event. Fundamentally

[2] For Chiron I used the code provided by the authors with some minor modifications, in case of Deepnano I have used my own implementation as the code provided by the authors had some issues with training on a GPU

the models differ with the way they take input, deepnano requires segmentation while raw signals can be fed in case of Chiron. The motivation behind to use CNN comes from the need to do automatic segmentation of local events as the process of segmentation requires additional knowledge on the way nanopore reads were captured which might not be always available. However this comes at a cost of increased number of parameters making training and inference a complex and time consuming procedure. However there exists a trade-off that CNN-RNN model has a significantly superior performance when compared to just RNN, indicating that the local features learned by the CNN are more relevant for the succeeding RNN when compared to the handcrafted features used in Deepnano. Although Deepnano is rather simplistic when compared to Chiron resulting in significantly faster training and testing times, it fails to generalize well with unseen data thereby severely limiting it's use cases. Chiron on the other hand generalizes remarkably well with regard to unseen data.

There are some fundamental limitations to both models discussed in this report. Firstly the results reported here are biased towards the data I have used in my experiments. In other words the conclusions are derived from a subset of E.Coli and Phage Lambda nanopore data and might not necessarily be applicable to all other data. Secondly both the models require some sort of labeling of raw data. In case of deepnano, segmentation is needed both for training and testing phases, while Chiron only requires labeling during training. Both segmentation and labeling are not trivial tasks, require additional information apart from the raw signals, which may not be easily available. Lastly the accuracy obtained from Chiron is nowhere close to the performance of Illumina sequencing technology and cannot be used to sequence new genomes in it's current form.

5. FUTURE WORK

Sequencing methods for nanopore data are essentially data driven as opposed to first and second generation methods which basically exploit the chemical and physical properties of DNA to obtain base sequences with very high accuracy. An interesting direction to look is to improve nanopore read efficiency by having electric signals which require very little or no post processing for obtaining base sequences. Another direction that can be looked upon is building a generative model of a sequencer where a distribution is obtained as opposed to a discriminative way of modeling, as is done currently in case of Chiron and Deepnano.

6. REFERENCES

[1] D. Branton, D. W. Deamer, A. Marziali, H. Bayley, S. A Benner, T. Butler, M. Di Ventra, S. Garaj, A. Hibbs, X. Huang, and *et.al*, "The potential and challenges of nanopore sequencing," in *Nanoscience And Technology: A Collection of Reviews from Nature Journals*, pp. 261–268. World Scientific, 2010.

[2] V. Boža, B. Brejová, and T. Vinař, "Deepnano: deep recurrent neural networks for base calling in minion nanopore reads," *PloS one*, 2017.

[3] M. David, L. J. Dursi, D. Yao, P. C Boutros, and J. T Simpson, "Nanocall: an open source basecaller for oxford nanopore sequencing data," *Bioinformatics*, vol. 33, no. 1, pp. 49–55, 2016.

[4] H. Teng, M. D. Cao, M. B. Hall, T. Duarte, S. Wang, and L. JM Coin, "Chiron: Translating nanopore raw signal directly into nucleotide sequence using deep learning," *GigaScience*, 2018.

[5] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[6] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen, and *et.al*, "Deep speech 2: End-to-end speech recognition in english and mandarin," in *International Conference on Machine Learning*, 2016, pp. 173–182.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

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

[9] H. Teng, M. D. Cao, M. B. Hall, T. Duarte, S. Wang, and L. JM Coin, "Supporting data for chiron: translating nanopore raw signal directly into nucleotide sequence using deep learning.," *Gigascience Database*, 2018.

[10] M. H. Stoiber, J. Quick, R. Egan, J. E. Lee, S. E. Celniker, R. Neely, N. Loman, L. Pennacchio, and J. B. Brown, "De novo identification of dna modifications enabled by genome-guided nanopore signal processing," *bioRxiv*, p. 094672, 2016.

[11] I. Sović, M. Šikić, A. Wilm, S. N. Fenlon, S. Chen, and N. Nagarajan, "Fast and sensitive mapping of nanopore sequencing reads with graphmap," *Nature communications*, vol. 7, pp. 11307, 2016.