MinCall — MinION end2end deep learning basecaller

Neven Miculini, Marko Ratkovi, and Mile Siki

Faculty of Electrical Engineering and Computing (FER), Zagreb, Croatia

Abstract. The Oxford Nanopore Technologies's MinION is the first portable DNA sequencing device. It's capable of producing long reads, over 100 kBp were reported, however, is has significantly higher error rate than other methods.
In this study, we created MinCall, an end2end basecaller model for the MinION. The model is based on deep learning and uses convolutional neural networks (CNN) in its implementation. For extra performances is uses cutting edge deep learning techniques and architectures, notably gated residual network, batch normalization and Connectionist Temporal Classification (CTC) loss.
The best performing 270 layers deep model achieves state-of-the-art 90.5% match rate on E.Coli dataset using R9 pore chemistry and 1D reads.

Keywords: Basecaller, MinION, R9, CNN, CTC, Next generation sequecing

1 Introduction

Oxford Nanopore Technology's MinION nanopore sequencing platform [Mikheyev and Tin(2014)] is the first portable DNA sequencing device. It's small weight, of only 90 grams, low capital cost, and long read length combined with high-throughput, real-time data analysis, and decent accuracy yield promising results in various applications. From clinical application such as monitoring infectious disease outbreaks [Judge et al(2015)Judge, Harris, Reuter, Parkhill, and Peacock] [Quick et al(2016)Quick, Loman, Duraffour, Simpson, Severi, Cowley, Bore, Koundouno, Dudas, Mikhail et al], characterizing structural variants in cancer [Norris et al(2016)Norris, Workman, Fan, Eshleman, and Timp] and even full human genome assembly [Jain et al(2017)Jain, Koren, Quick, Rand, Sasani, Tyson, Beggs, Dilthey, Fiddes, Malla et al].

Although MinION is able to produce long reads, they have a high sequencing error rate. This has been somewhat alleviated with new R9 pore model, replacing older R7 ones. In this paper, we show that this error rate can be reduced by replacing the default base caller provided by the manufacturer with a properly trained neural network model. In the future new R9.5 chemistry and 1D$\hat{2}$ reads supersede current models.

In the MinION device, single-stranded DNA fragments move through nanopores, which causes drops in the electric current. The electric current is measured at each pore several thousand times per second, 4000 times exactly in our dataset. The electric current depends mostly on the context of several DNA bases passing through the pore at the time of measurement. As the DNA moves through the pore, the context shifts, and the electric current changes.

A MinION device typically yields reads several thousand bases long; reads as long as 100,000 bp have been reported. 1D read has usually error rate about 10%, though it varies across different genomes, human one being lower accuracy than E.Coli one. (Check details!!, What is the error rate, how it's defined???) After parameter training, base calling can be performed by running the Viterbi algorithm, which will result in the sequence of states with the highest likelihood.

The exact error rate metric is unreliable since multiple pipeline tools could be the issue. First the sample is prepared, hopefully, uncontaminated and matching reference genome as close as

possible, then sequenced using the MinION device obtaining raw data. Next, our model (or other groups ones) come along, basecall the sequence. To evaluate error rate metric basecalled read is aligned to the reference genome using aligners with their own errors/biases, mostly commonly used BWA-MEM [Li(2013)] and Graphmap [Sović et al(2016)Sović, Šikić, Wilm, Fenlon, Chen, and Nagarajan].

2 Background

3 Sequencing overview

Conceptually, the MinION sequencer works as follows. First, DNA is sheared into fragments of 820 kbp and adapters are ligated to either end of the fragments. The resulting DNA fragments pass through a protein embedded in a membrane via a nanometre-sized channel, a nanopore. A single DNA strand passes through the pore. Optionally, hairpin protein adapter can merge two DNA strands, allowing both template and complement read passing through the nanopore sequentially for more accurate reads. This technique is referred as 2D reads, while we focus on 1D reads containing only template DNA and no hairpin adapter.

Electrical current runs through the nanopore and exact nucleotides context within influences the nanopore's resistance. This resistive effect is our sensor data, that is the current fluctuations as DNA passes though the pore. The nanopore is 6 nucleotides wide, and many models use this information in their approaches, while we're opted out of this technicality.

4 Basecalling

The core of the decoding process is the basecalling step. Official basecaller is Metrichor, previously performed in the cloud before being open-sourced under name Nanonet. (Citation, whom?)

Earlier models were (Hidden Markov model) HMM-based where hidden state modeled DNA sequence of length 6 (6-mer) in the nanopore. Pore models were used in computing emission probabilities. [Loman et al(2015)Loman, Quick, and Simpson, Schreiber and Karplus(2015), Szalay and Golovchenko(2015), Timp et al(2012)Timp, Comer, and Aksimentiev] and the recent open source HMM-based basecaller Nanocall [David et al(2016)David, Dursi, Yao, Boutros, and Simpson].

Recent models opted using (Recurrent neural network) RNN, notably, DeepNano [Boža et al(2016)Boža, Brejová, and Vinař] and recently open sourced official Nanonet from ONT.

5 Method

Instead of opting for the traditional path using HMM or RNN we tried using CNN (Convolutional neural networks) [LeCun et al(1998)LeCun, Bottou, Bengio, and Haffner], that is their residual version [He et al(2016)He, Zhang, Ren, and Sun]. We opted out for gated residual network variant [Savarese(2016)]. For loss, we used CTC (Connectionist temporal classification) [Graves et al(2006)Graves, Fernández, Gomez, and Schmidhuber] between basecalled and the target sequence. The implementation used is open source warp-ctc [Amodei et al(2015)Amodei, Anubhai, Battenberg, Case, Casper, Catanzaro, Chen, Chrzanowski, Coates, Diamos, Elsen, Engel, Fan, Fougner, Han, Hannun, Jun, LeGresley, Lin, Narang, Ng, Ozair, Prenger, Raiman, Satheesh, Seetapun, Sengupta, Wang, Wang, Wang, Xiao, Yogatama, Zhan, and Zhu]. Main computation framework is tensorflow [Abadi et al(2015)Abadi, Agarwal, Barham, Brevdo, Chen, Citro, Corrado, Davis, Dean, Devin, Ghemawat, Goodfellow, Harp, Irving, Isard, Jia, Jozefowicz, Kaiser,

Kudlur, Levenberg, Mané, Monga, Moore, Murray, Olah, Schuster, Shlens, Steiner, Sutskever, Talwar, Tucker, Vanhoucke, Vasudevan, Viégas, Vinyals, Warden, Wattenberg, Wicke, Yu, and Zheng]

6 Data preprocessing

Dataset was obtained from (Ask somebody, Marko doesn't know!). In this research, models were trained on the E. Coli K-12 strand. (Should I cite someone for strand). Data was split into train and test subset, such that aligned reads map to different reference genome parts. For initial model data bootstrapping metrichorn was used, version (Ask Someone?).

The fast5 input training files were further split into smaller training blocks, consisting of fixed block size on raw signal. For each block target sequence, basecalled data is used in following way.

We're using basecalled knowledge which tells us on each raw read part which 6-mer were currently in the nanopore. Using this data, we get the intermediate target sequence for each block. To correct for model errors, we use aligned information, that is reference genome and alignment data (cigar string) to correct that information, yielding finalized target sequence. For further performance, we skip first and last training block, since most error aggregate on edges.

Adjacent matching bases are separated with surrogate one, for example, AAA -¿ AA'A for reasons described in the following section.

7 Residual arhitecture

We use gated residual network, that is layer function is like this: $f(x) = kg(x) + x$ Where x is input layer and k constant learnt during training. Specifically, g used is multiple, 1–3 times in our experiments, Relu-BatchNorm-CNN layers.

8 CTC

This section gives a brief overview over CTC, for further detail and in depth explanation we recommend original paper [Graves et al(2006)Graves, Fernández, Gomez, and Schmidhuber] or searching for contemporary blog posts.

In CTC we have target sequecne $\mathbf{l}$ consisting from symbols of alphabet Σ. Our models outputs $\mathbf{x}$, and CTC loss maximizes $p(\mathbf{l}|\mathbf{x})$.

$\mathbf{x}$ is consistend of n independent discrete random variables, X_i over domain $\Sigma \cup$Blank. Path p through $\mathbf{x}$ is assigning for each X_i single value. It's log probabity is $\log P(p) = \sum_i \log P(X_i = x_i)$ where x_is define the path.

We further have merging operator on path, merge$(x_1, x_2, \ldots, x_n)$ which merges together adjacent equivalent elements. For example merge$(AAAGC) = AGC$.

Then $p(\mathbf{l}|\mathbf{x}) = \sum_{p \in \text{paths}(\mathbf{x}, \text{merged}(p)=\mathbf{l}} P(p)$ or less formally, $P(\mathbf{l}|\mathbf{x})$ is sum of all path probabilites which when merged give $\mathbf{l}$

In out concrete x application $\Sigma = \{$A G T C A' G' T' A'$\}$

For decoding, we use beam search decoder, with beam width 100.

9 Results

9.1 Final model

Best performing model used has 270 total layers, divided into 3 90-layer blocks. Between each 90 layers blocks, MaxPool layer is inserted with the receptive width of 2 and stride 2, to ease computation effort and add precision.

Each block consists of 30 gated residual layers, each residual layer composed of 3 sequential Relu-Batch Normalization-Conv1D layers. Each convolutional layer uses the receptive width of 3 with 64 channels as output throughout the model.

9.2 Performance tables

As described in the previous section, the model was tested on E.Coli test set and compared to open source Nanonet and Albacore. Albacore doesn't have specific R9 chemistry mode, thus R9.4 was used instead which explains its lower performance on this task. Mean CIGAR operation are in table1 and KDE Match rate plot is in figure 9.2.

	Deletion rate	Error rate	Insertion rate	Match rate	Mismatch rate	Read length
albacore	0.060	0.194	0.070	0.867	0.063	9843
nanonet	0.088	0.190	0.040	0.897	0.062	5029
mincall_m270	0.077	0.172	0.040	0.905	0.056	9378

Table 1. Mean performance metrics on E.Coli dataset, 5k sample

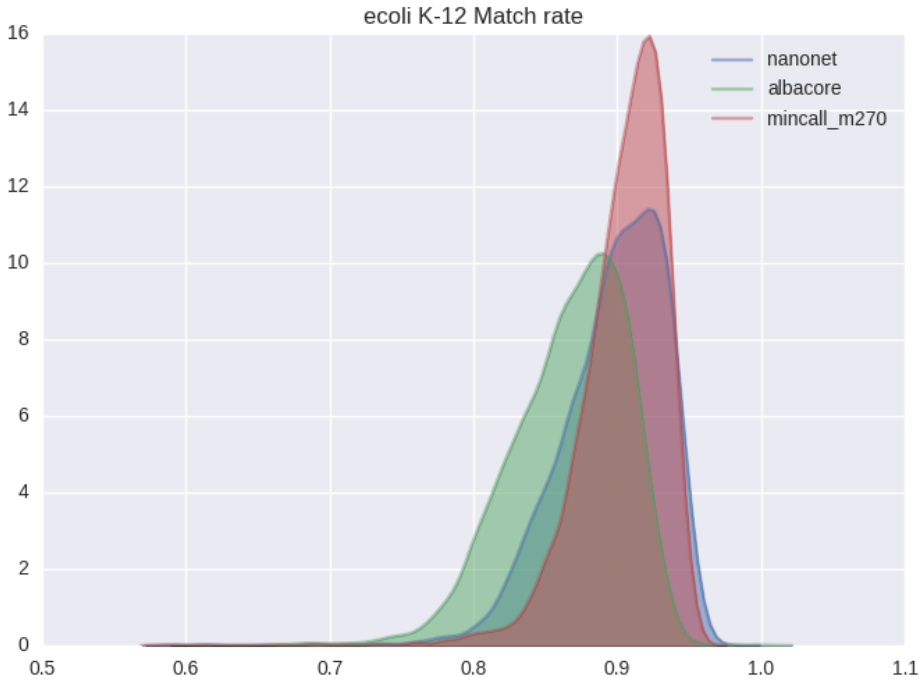
	Base pairs/second
albacore	38000
nanonet	Slow
mincall_m270	3774

Table 2. Speed

10 Conclusion and further work

This model used advance state-of-the-art gated residual convolutional neural network, with top models having 270 layers and over 3M parameters, yet improvements over Metrichorn baseline are marginal. As the conclusion, it might be that we've reached Bayesian error rate for R9 chemistry. Furthermore, R9.5 and 1D$\hat{2}$ reads are under development which shall yield this paper's result obsolete quite soon, yet underlying code developed could easily be adjusted and trained on new data.

Unlike Nanonet which uses custom OpenCL kernels or Albacore — a novel ONT basecaller as of May 2017 lacking GPU support, this work used world-class computational framework tensorflow with highly optimized kernels and large development community. Therefore resulting paper's effect is showcasing Residual CNN approach or pure CNN approach with CTC loss is marginally better than already established basecaller and providing code in the contemporary framework.



11 Acknowledgments

First and foremost I'd like to thank my mentor izv. prof. dr. sc. Mile Siki for setting up the problem, guiding us through its beginnings and providing helpful advice for its completion. Next Marko Ratkovic for his not only insightful conversations, but also meaningful code contributions.

Finally, I'm thanking various other people whose code, tools and advice I've used in completing this paper: Fran Jurii, Ana Marija Selak, Ivan Sovi and Martin oi.

References

Abadi et al(2015)Abadi, Agarwal, Barham, Brevdo, Chen, Citro, Corrado, Davis, Dean, Devin, Ghemawat, Goodfellow, Harp, Irv
Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado GS, Davis A, Dean J, Devin M, Ghemawat S, Goodfellow I, Harp A, Irving G, Isard M, Jia Y, Jozefowicz R, Kaiser L, Kudlur M, Levenberg J, Mané D, Monga R, Moore S, Murray D, Olah C, Schuster M, Shlens J, Steiner B, Sutskever I, Talwar K, Tucker P, Vanhoucke V, Vasudevan V, Viégas F, Vinyals O, Warden P, Wattenberg M, Wicke M, Yu Y, Zheng X (2015) TensorFlow: Large-scale machine learning on heterogeneous systems. URL `http://tensorflow.org/`, software available from tensorflow.org

Amodei et al(2015)Amodei, Anubhai, Battenberg, Case, Casper, Catanzaro, Chen, Chrzanowski, Coates, Diamos, Elsen, Engel, F
Amodei D, Anubhai R, Battenberg E, Case C, Casper J, Catanzaro B, Chen J, Chrzanowski M, Coates A, Diamos G, Elsen E, Engel J, Fan L, Fougner C, Han T, Hannun AY, Jun B, LeGresley P, Lin L, Narang S, Ng AY, Ozair S, Prenger R, Raiman J, Satheesh S, Seetapun D, Sengupta S, Wang Y, Wang Z, Wang C, Xiao B, Yogatama D, Zhan J, Zhu Z (2015) Deep speech 2: End-to-end speech recognition in english and mandarin. CoRR abs/1512.02595, URL `http://arxiv.org/abs/1512.02595`

Boža et al(2016)Boža, Brejová, and Vinař. Boža V, Brejová B, Vinař T (2016) DeepNano: Deep Recurrent Neural Networks for Base Calling in MinION Nanopore Reads. ArXiv e-prints 1603.09195

David et al(2016)David, Dursi, Yao, Boutros, and Simpson. David M, Dursi LJ, Yao D, Boutros PC, Simpson JT (2016) Nanocall: an open source basecaller for oxford nanopore sequencing data. Bioinformatics p btw569

Graves et al(2006)Graves, Fernández, Gomez, and Schmidhuber. Graves A, Fernández S, Gomez F, Schmidhuber J (2006) Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In: Proceedings of the 23rd international conference on Machine learning, ACM, pp 369–376

He et al(2016)He, Zhang, Ren, and Sun. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 770–778

Jain et al(2017)Jain, Koren, Quick, Rand, Sasani, Tyson, Beggs, Dilthey, Fiddes, Malla et al. Jain M, Koren S, Quick J, Rand AC, Sasani TA, Tyson JR, Beggs AD, Dilthey AT, Fiddes IT, Malla S, et al (2017) Nanopore sequencing and assembly of a human genome with ultra-long reads. bioRxiv p 128835

Judge et al(2015)Judge, Harris, Reuter, Parkhill, and Peacock. Judge K, Harris SR, Reuter S, Parkhill J, Peacock SJ (2015) Early insights into the potential of the oxford nanopore minion for the detection of antimicrobial resistance genes. Journal of Antimicrobial Chemotherapy 70(10):2775–2778

LeCun et al(1998)LeCun, Bottou, Bengio, and Haffner. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proceedings of the IEEE 86(11):2278–2324

Li(2013). Li H (2013) Aligning sequence reads, clone sequences and assembly contigs with bwa-mem. arXiv preprint arXiv:13033997

Loman et al(2015)Loman, Quick, and Simpson. Loman NJ, Quick J, Simpson JT (2015) A complete bacterial genome assembled de novo using only nanopore sequencing data. Nature methods 12(8):733–735

Mikheyev and Tin(2014). Mikheyev AS, Tin MM (2014) A first look at the oxford nanopore minion sequencer. Molecular ecology resources 14(6):1097–1102

Norris et al(2016)Norris, Workman, Fan, Eshleman, and Timp. Norris AL, Workman RE, Fan Y, Eshleman JR, Timp W (2016) Nanopore sequencing detects structural variants in cancer. Cancer biology & therapy 17(3):246–253

Quick et al(2016)Quick, Loman, Duraffour, Simpson, Severi, Cowley, Bore, Koundouno, Dudas, Mikhail et al. Quick J, Loman NJ, Duraffour S, Simpson JT, Severi E, Cowley L, Bore JA, Koundouno R, Dudas G, Mikhail A, et al (2016) Real-time, portable genome sequencing for ebola surveillance. Nature 530(7589):228–232

Savarese(2016). Savarese PH (2016) Learning identity mappings with residual gates. arXiv preprint arXiv:161101260

Schreiber and Karplus(2015). Schreiber J, Karplus K (2015) Analysis of nanopore data using hidden markov models. Bioinformatics p btv046

Sović et al(2016)Sović, Šikić, Wilm, Fenlon, Chen, and Nagarajan. Sović I, Šikić M, Wilm A, Fenlon SN, Chen S, Nagarajan N (2016) Fast and sensitive mapping of nanopore sequencing reads with graphmap. Nature communications 7

Szalay and Golovchenko(2015). Szalay T, Golovchenko JA (2015) De novo sequencing and variant calling with nanopores using poreseq. Nature biotechnology 33(10):1087–1091

Timp et al(2012)Timp, Comer, and Aksimentiev. Timp W, Comer J, Aksimentiev A (2012) Dna basecalling from a nanopore using a viterbi algorithm. Biophysical journal 102(10):L37–L39