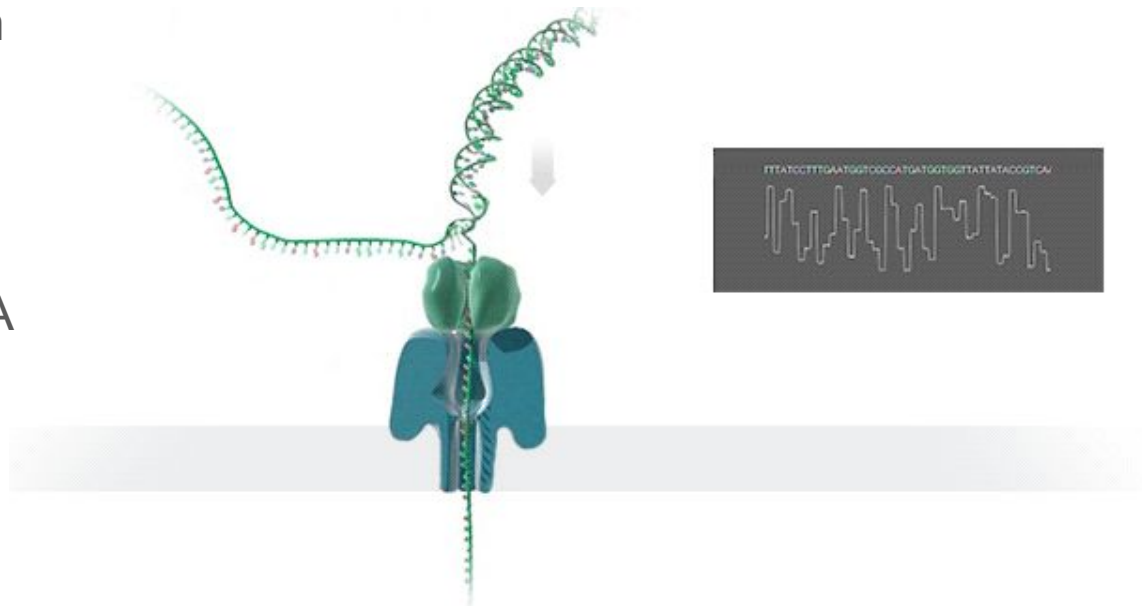
BasecRAWller: Streaming Nanopore Basecalling Directly from Raw Signal

Marcus Stoiber and James Brown
Presented by Sam Kovaka

Nanopore Sequencing

- Sequences DNA/RNA by measuring changes in ionic current as a nucleotide strand passes through a pore
- Each k-mer (k = 4-6) has an expected current value
- Basecalling: translating nanopore data acquisition (DAC) values into DNA/RNA sequence



Nanopore Basecalling (background)

Typically a multi-step process

- Segmentation - rough mapping of DAC values to events (k-mers)
- Normalization - scale event levels to account for pore differences
- Basecalling - translate event levels to bases (usually w/ HMM or neural net)

Previous methods require full read for normalization and basecalling steps

Segmentation (background)

- Collapsing multiple (average of 9) DACs into single events
- Ideally one event represents one k-mer, but often multiple events measure same k-mer or multiple k-mers are in one event
- Typically done before basecalling
- Major development in BasecRAWller is calling events before segmentation

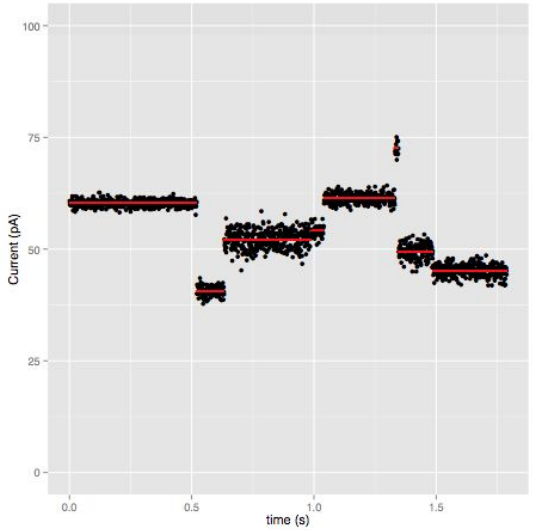
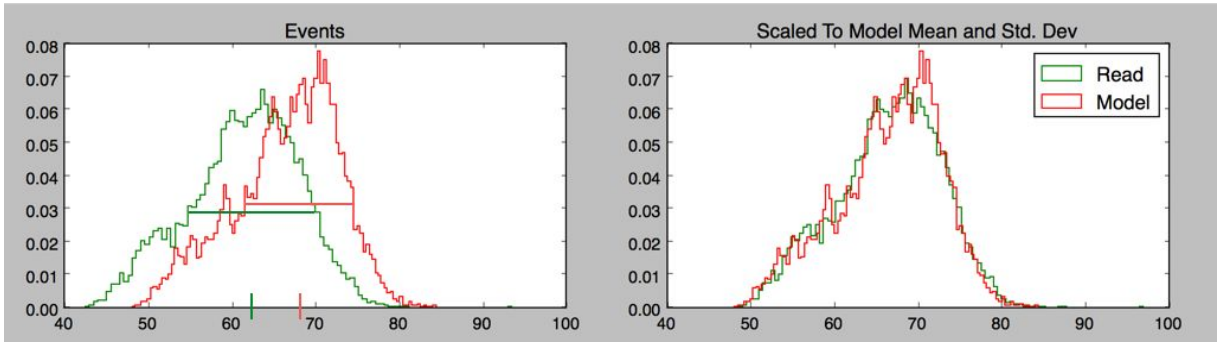


Figure S1: Simulated ideal signal data to illustrate the data that is input into our model. The black points are sampled current levels at a given time. The red lines are the events detected by a feature detection algorithm, which partitions the samples into discrete segments.

Normalization (background)

- Small differences in pore conditions cause DACs to be consistently scaled and shifted from expected values
- Normalization can be performed quickly at the cost of accuracy (method of moments) or with more sophisticated methods (expectation-maximization)
- All previous methods require full read to compute scaling values



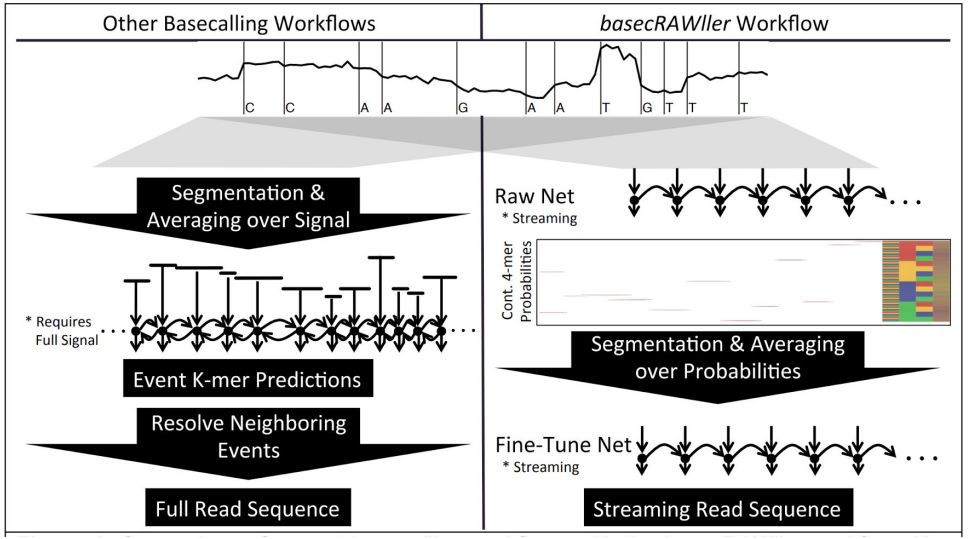
http://simpsonlab.github.io/2015/12/18/kdtree-mapping/

Basecalling (background)

- Translation of scaled events to bases
- Earlier methods tend to use hidden markov models (HMMs)
  - Nanocall, old ONT basecaller
- More recent approaches use neural networks
  - Deep nano, new ONT basecaller
- All methods are time consuming and require full read

BasecRAWller

- First streaming basecaller
  - Doesn’t require full read and can call bases faster than they are sequenced
- Calls individual DACs rather than events, then segments
- Composed of two unidirectional neural networks
  - Raw net - calls observations
  - Fine-tune net - resolves segmented observations
- Segments between neural nets
- Normalization doesn’t require full read

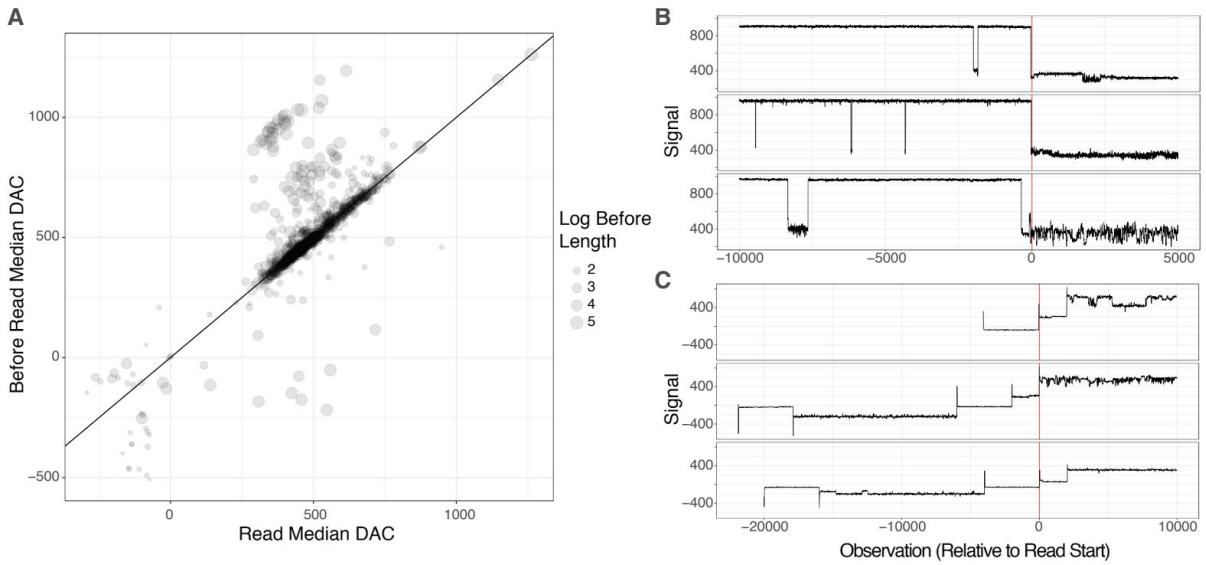


Why Streaming?

- Nanopore sequencers have “read until” capabilities - can stop sequencing a read and shift to another if it’s decided that read is not useful
- A single read can take as much as 30 minutes to sequence
- Can’t decide if a read is useful until it’s basecalled, so doing so before the read is finished is crucial

BasecRAWller Normalization

- Uses signal before the strand is in the pore to normalize future DACs
- Median signal before read correlates well (r = 0.76) with median signal within
- Finding where the read begins is not trivial
  - Should be where signal drops, but random noise can create false positives

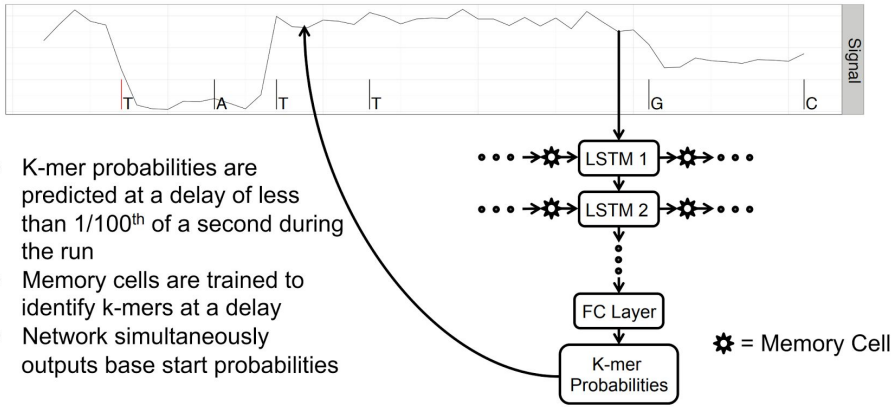


Raw Neural Net

Unidirectional recurrent neural network (RNN)

Input: one normalized DAC value, previous raw net output

Output: probability DAC represents each 4-mer (256 probabilities), probability that the nanopore has moved to a new base since last input (used for segmentation)



- K-mer probabilities are predicted at a delay of less than 1/100th of a second during the run
- Memory cells are trained to identify k-mers at a delay
- Network simultaneously outputs base start probabilities

BasecRAWller Segmentation

- Uses probability each position is a new base output by the raw net
- Uses peaks in these probabilities as segmentation points
  - Peaks must be above some value determined in training
- Proposes two other segmentation methods:
  - Hellinger distance - measures distance between multinomial distance. Distance should shift significantly between bases
  - Entropy distance - shifts in entropy should correspond to base transitions
- Probability for each 4-mer is then averaged within each segment

Fine-Tuning Neural Net

Input: 4-mer probabilities of a single segment, previous base calls

Output: zero, one, or two basecalls

Same basic structure as raw net

Ideally would output single base for each segment, but could output more or less if the segmenter made a mistake.

Training

- Uses real nanopore reads, basecalled by ONT's basecaller, to train the neural networks
- Error corrects reads using nanoraw
  - Aligns reads to reference, corrects mismatches, and outputs true mapping from DACs to basecalls
- Trained on 3,740 reads starting at random locations

Results

- Trained and tested on two datasets using different procedures
  - E. coli: FLOMIN-105, 250bps, 1D reads, R9 chemistry
  - Human: 1D ligation, 450bps, 2D reads, R9.4 chemistry
- Also tested on E. coli dataset not used for training
- Consistently did worse than ONT, but basecalled ~2x faster than the reads were sequenced
- Human training is “more robust”