A Deep Learning Solution to Advance Blood Diagnostics of Cancers

Zaid Nabulsi, Shuvam Chakraborty, Vineet Kosaraju

Background

Blood diagnostics of cancer present a golden opportunity to advance the state of cancer treatment. However, diagnosing cancer through the blood is an extremely difficult task that is analogous to finding a needle in a haystack, but is possible due to the presence of circulating tumor DNA. To aid the development of blood cancer diagnostics, we will build a binary classifier using a deep learning model that will accurately ascertain whether a single non-reference base is human introduced, or whether it may be an indicator of disease.

Features

Raw data was obtained from The Alizadeh Lab in the Stanford School of Medicine, which has collected full genome sequences from healthy patients and patients with cancers. We obtained DNA sequences from over 300 different patients and extracted relevant features from the raw sequence data. A description of these features can be found below.

Feature Name	Feature Description
Allele Frequency	General proportion of chromosomes containing specific allele.
Barcode Family	# of PCR duplicates generated
Base Change	A constant (0-11) representing what base change is observed (eg A to G)
Duplex	Binary feature for whether the fragment comes from a duplex molecule
Read 1 / Read 2	Binary feature representing whether the read is the Watson or the Crick strand.
Plus / Minus	Binary feature representing whether the strand is a plus or minus strand
Position on Read	A number between 0 (start) and 1 (end) for where on the read the base was.
Number Non Reference Bases	Number of non reference bases on the read, including the current base
Base Quality	The PHRED base quality of the base
Mapping Quality	The mapping quality of the read
Fragment Length	The length of the fragment
P-Value	The polishing p-value for the given base, generated from a background database of healthy samples
Polish Normally	Whether this base would get polished out or not, strictly based on p-value
Type of Cancer	Which cancer type the base is from

Dataset Overview

Our final dataset included roughly 30 million training examples. Each training example represents a single non-reference base found in a patient's DNA sequence. Due to the nature of the problem, the distribution of our data is skewed. Approximately 86.3 percent of our training examples are labelled class 0 (indicating human processing error), and the remaining 13.7 percent labelled class 1 (biological). We discuss how we addressed this problem in the methods section. The data was split into a 90/5/5 train/dev/test split.

Baseline Evaluation

For baseline performance, we used three models:

•Model null distribution of human-introduced error and classify with p values using a statistical framework

•Basic logistic regression.

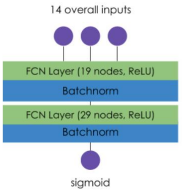
•Neural network with 1 hidden layer and an output layer.

Models were evaluated with precision, recall, accuracy, and F1 score on the oversampled train and dev sets (see methods/loss). This allowed us to bypass the data skew problem.

Metric	P-Value Feature	Logistic Regression	2 Layer NN
Train Accuracy	0.883245	0.90412	0.91643
Train Precision	0.456595	0.52361	0.53461
Train Recall	0.504761	0.50765	0.51089
Train F1 Score	0.479472	0.51550	0.52248
Dev Accuracy	0.883284	0.90421	0.91632
Dev Precision	0.456390	0.52351	0.53426
Dev Recall	0.505119	0.50752	0.51078
Dev F1 Score	0.479520	0.51391	0.52225

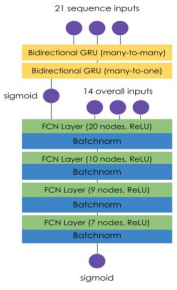
Methods

Model 1("Deep Net")



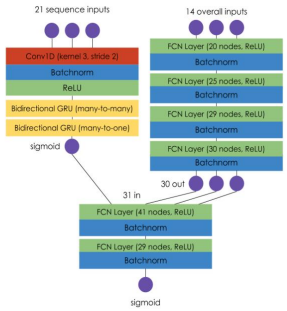
- 512 batch size
- Learning rate 1e-4 for 5 epochs

Model 2("TwoNet")



- 512 batch size
- Learning rate 1e-4 for 10 epochs

Model 3("ThreeNet")



- 512 batch size
- Learning rate (with decay) 2e-4 for 15 epochs

Hyperparameters

For all the models we tried out, we tuned a variety of hyperparameters to improve each model's performance. Some of the many hyperparameters we experimented with include:

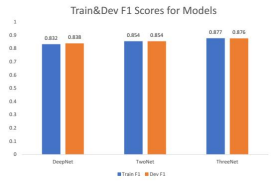
- size/number of layers/nodes
- Learning rate and # of epochs
- Mini-batch size
- GRU vs RNN vs LSTM
- Activation functions

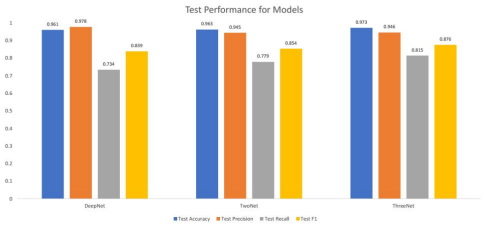
Loss

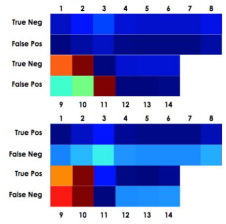
To fix data skew, we oversampled class 1 labels in the training set. We used cross-entropy loss to train the models.

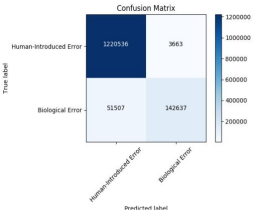
Results & Error Analysis

Model / Metric	Deep NN	TwoNet	ThreeNet
Train Accuracy	0.961139	0.963553	0.972561
Train Precision	0.976161	0.944275	0.947189
Train Recall	0.734467	0.780178	0.815891
Dev Accuracy	0.961102	0.963459	0.972189
Dev Precision	0.974962	0.943015	0.946456
Dev Recall	0.734696	0.780194	0.814567









References

Song, Li, et al. "Lighter: fast and memory-Efficient sequencing error correction without counting." *Genome Biology*, BioMed Central, 15 Nov. 2014, genomebiology.biomedcentral.com/articles/10.1186/s13059-014-0509-9.

Kircher, Martin, et al. "Improved base calling for the Illumina Genome Analyzer using machine learning strategies." *Genome Biology*, BioMed, 14 Aug. 2009, genomebiology.biomedcentral.com/articles/10.1186/gb-2009-10-8-r83.

Rajpurkar, Pranav, et al. "Cardiologist-Level Arrhythmia Detection with Convolutional Neural Networks." *Cardiologist-Level Arrhythmia Detection with Convolutional Neural Networks*, 6 July 2017, arxiv.org/abs/1707.01836.

Libbrecht, Maxwell W., and William Stafford Noble. "Machine learning applications in genetics and genomics." *Nature Magazine*, Nature, www.nature.com/articles/nrg3920.

Conclusions

This was a challenging project due to the nature and quantity of the data as well as the complexity of the models we tried. Nevertheless, we were able to attain stronger accuracy and F1 numbers while improving model complexity. While it may be possible to improve our model through further training and hyperparameter search, our results so far give us hope that this method can eventually be used in the medical field to improve the prospects of blood cancer diagnostics.