Introduction to Karas & Impact of Genomic Context on Variant Calling

Arkarachai Fungtammasan (Chai)
Steve Osazuwa
Naina Thangaraj
Jason Chin

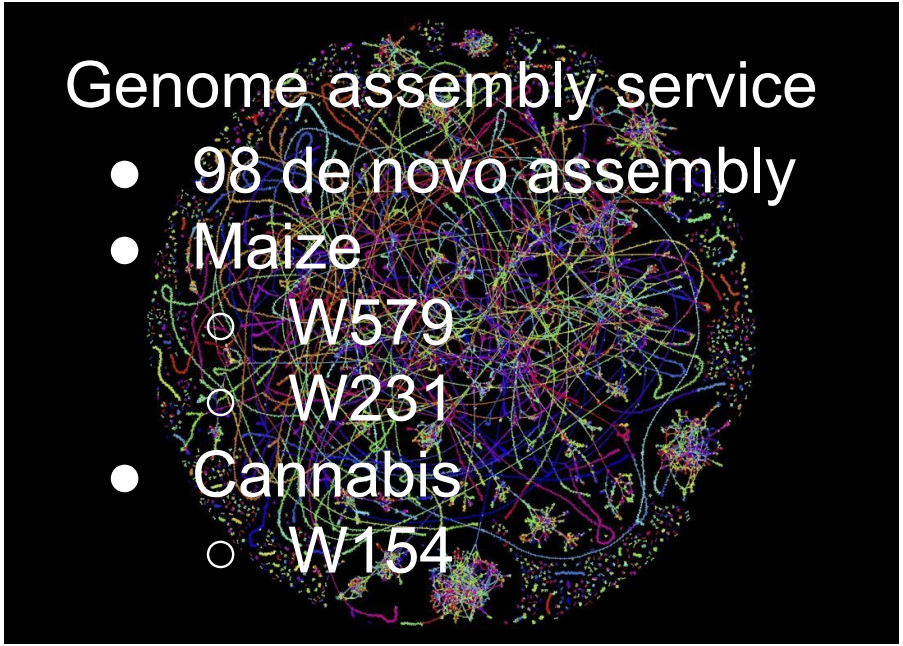
DNAnexus®
The Global Network For Genomics™

About myself

About myself

Booth 227

DNAnexus

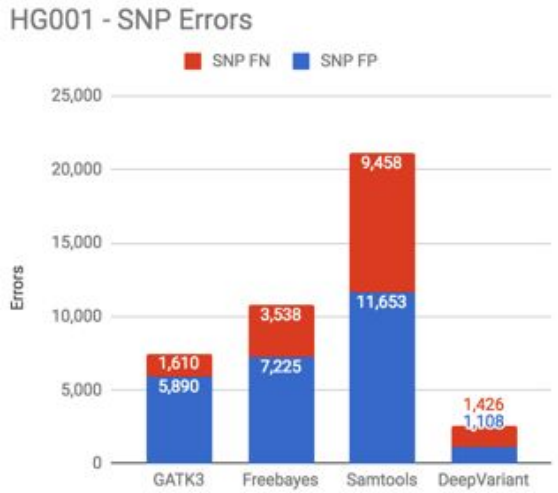


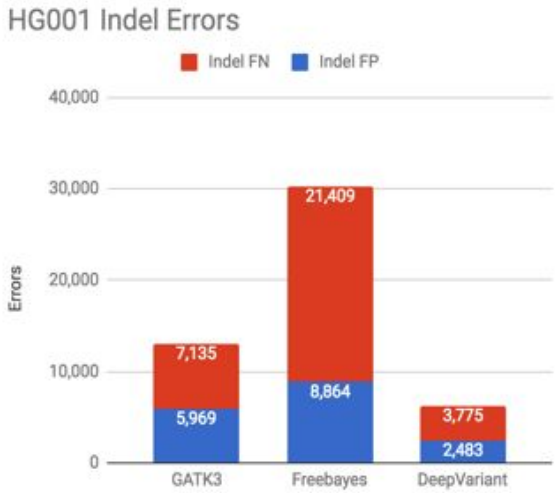
Science frontier program

- 10% of time to work on cutting edge of science and contribute back to scientific community
- Making blog post https://blog.dnanexus.com, preprint, or give a talk at the conference

Science frontier program

EVALUATING DEEPVARIANT: A NEW DEEP LEARNING VARIANT CALLER FROM THE GOOGLE BRAIN TEAM





Outlines

- What is Keras? And why?
- How to get start?
- Basic Keras code structure
- Real world application: Impact of genomic context in Variant calling

Outlines

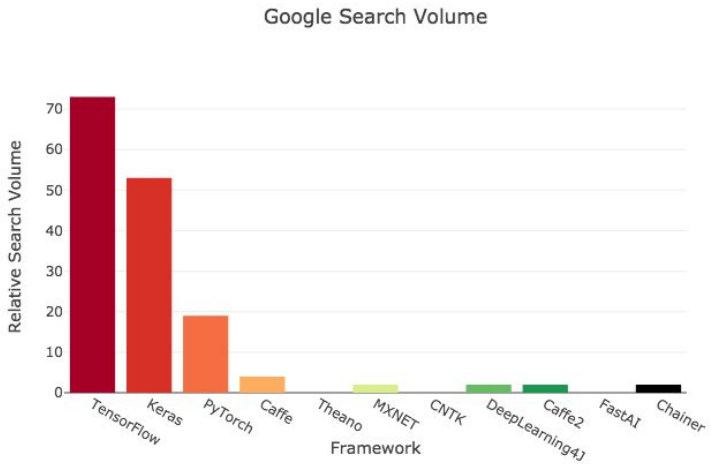
- What is Keras? And why?
- How to get start?
- Basic Keras code structure
- Real world application: Impact of genomic context in Variant calling

Keras (care-ras)

- High level Python library/API for deep learning run on top of TensorFlow, CNTK, and Theano
- Design with user friendliness, modularity, and extensibility principles
- Support common stuffs: Model
  - Model: CNN, RNN, combination
  - Hardware: CPU, GPU, TPU, Spark

Why Keras ?

- Easy to learn and code. “Keras is an API designed for human beings, not machines.”
- Popularity



How do you get start?

- https://keras.io
  - Install backend
  - pip install keras
- https://www.tensorflow.org/tutorials/
  - Install tensorflow
  - from tensorflow import keras
- Kaggle online kernel

Keras code structure

1 Model

```
def keras_dna_model():

    model = Sequential()

    model.add(Flatten(input_shape=(4, 12)))

    model.add(Dense(48, activation='relu'))

    model.add(Dense(1, activation='sigmoid'))

    return model
```

Picture from: flicker.com

Keras code structure

1 Model

```
def keras_dna_model(input_shape):
    X_input = Input(input_shape)
    X = Flatten()(X_input)
    X = Dense(12, activation='relu', name='n1')(X)
    X = Dense(1, activation='sigmoid', name='n2')(X)
    model = Model(inputs=X_input, outputs=X, name='keras_dna_model')
    return model
```

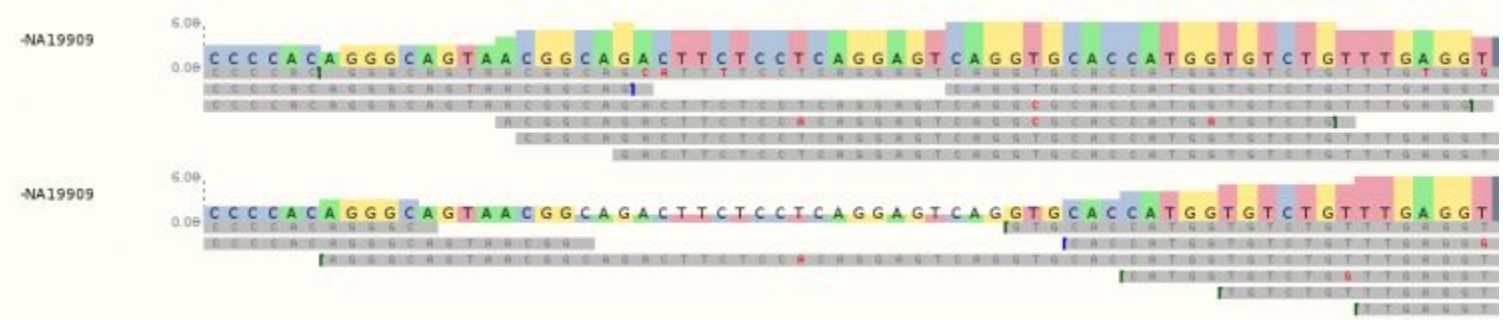
Picture from: flicker.com

Keras code structure

2 Run model

```
model = keras_dna_model(input_dimension)

model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])

model.fit(X_train, Y_train, epochs=2, batch_size=100)



model.predict(X_test)

model.evaluate(X_test, Y_test, batch_size=20168, verbose=1)
```

Impact of genomic context in variant calling



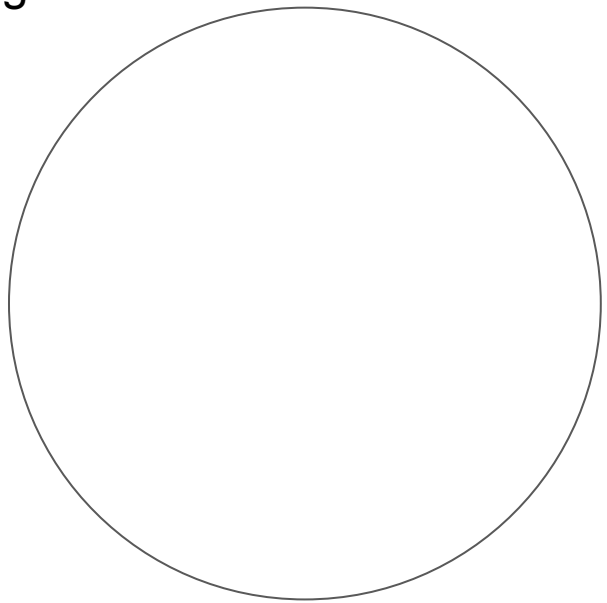
Picture from: EMBL

Problem formulation

Feature		Response
ATTCGACG	→	1
AAAATCCT	→	0

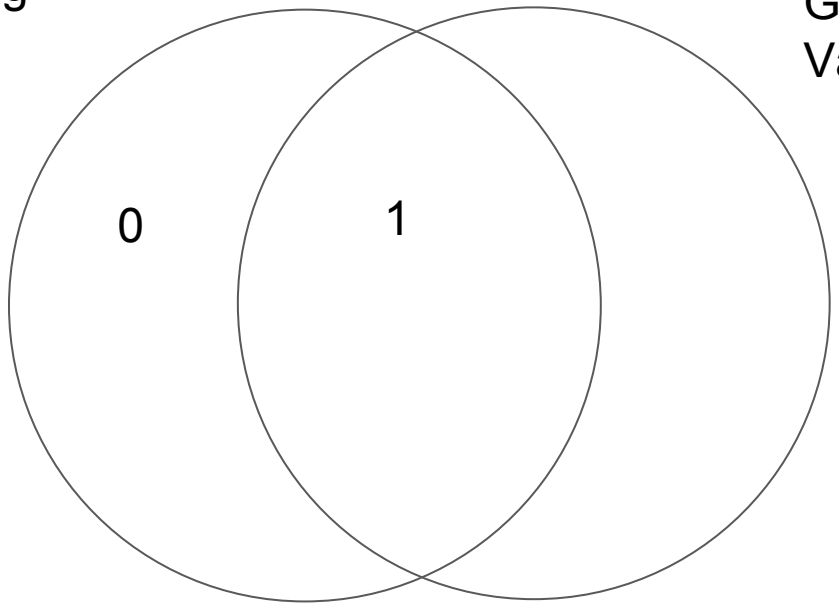
Data

Indel DeepVariant calling



Data

Indel DeepVariant calling



GIAB
Variant truth set

Why deep learning?

	feature1	feature2	feature3	feature4	response
1	5	2	1	-0.2	1
2	2	-2	1	5	0
3	9	-3	0	8.2	0
4	0	-2	1	0	1

Why deep learning?

	feature1	feature2	feature3	feature4	response
1	5	2	1	-0.2	1
2	2	-2	1	5	0
3	9	-3	0	8.2	0
4	0	-2	1	0	1

```
A T C A G C A T


1 0 0 1 0 0 1 0
0 0 1 0 0 1 0 0
0 0 0 0 1 0 0 0
0 1 0 0 0 0 0 1
```

Notebook demo

Caveats

- Size of context may not be optimized
- DeepVariant probably learn a lot about genomic context by itself

Next Step

- Production scale of parameter experiment using Papermill https://papermill.readthedocs.io/en/latest/
- Experiment on impact of context on conventional variant calling

Thanks to

Steve Osazuwa, Naina Thangaraj, Jason Chin

Jason Williams and Nirav Merchant

https://github.com/Arkarachai/pag2019_demo_keras_for_genomics

chai@dnanexus.com

DNAnexus®
The Global Network For Genomics™