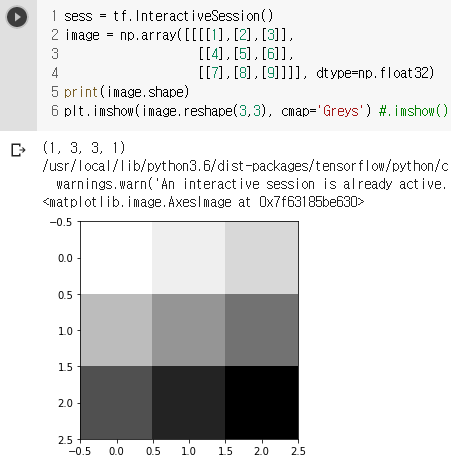
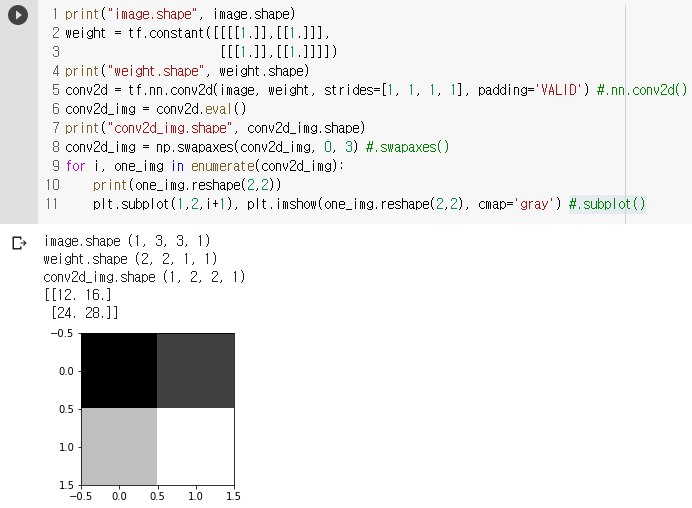
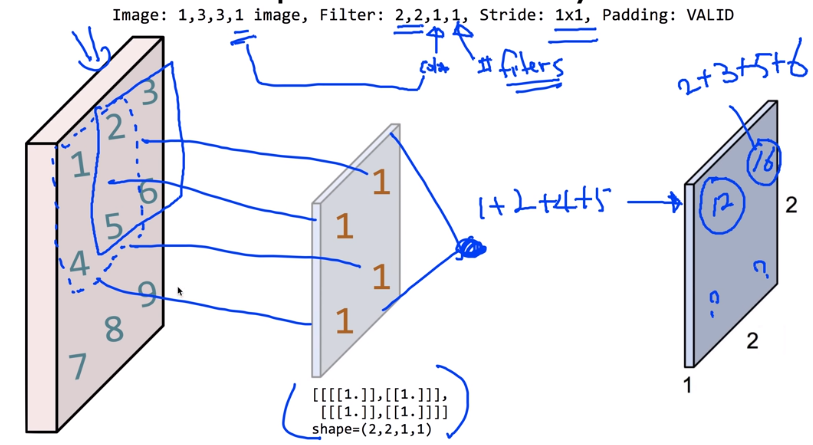
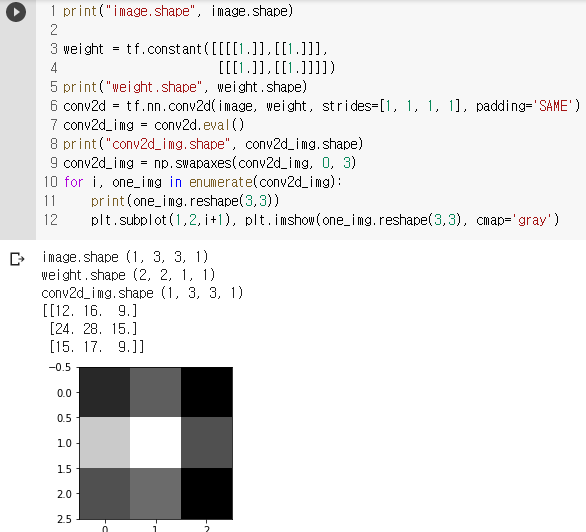
**ML\_Lab 11-1.**

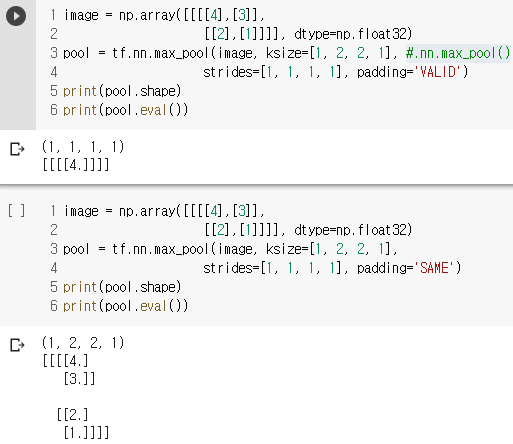
**Test.**







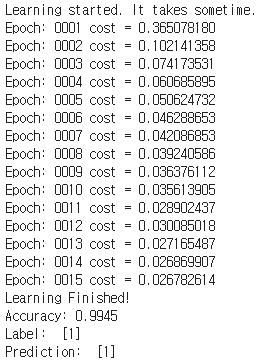




**ML\_Lab 11-2. 3.**

//Colab에서 Tensorflow 1.15.2를 불러와 실행할 수 있다. 시간이 상당히 많이 소요된다.

//이전에 GitHub, 강의 사진으로 대체했던 Lab도 1.15.2를 불러와 해결할 수 있었다.

//실제로 1.15.2를 불러와 mnist\_cnn의 결과가 나왔다.

<https://raw.githubusercontent.com/hunkim/DeepLearningZeroToAll/master/lab-11-1-mnist_cnn.py>

<https://raw.githubusercontent.com/hunkim/DeepLearningZeroToAll/master/lab-11-2-mnist_deep_cnn.py>

<https://raw.githubusercontent.com/hunkim/DeepLearningZeroToAll/master/lab-11-3-mnist_cnn_class.py>

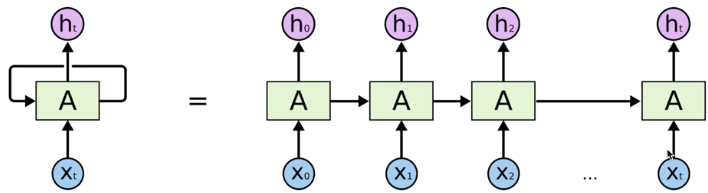
<https://github.com/hunkim/DeepLearningZeroToAll/blob/master/lab-11-4-mnist_cnn_layers.py>

<https://github.com/hunkim/DeepLearningZeroToAll/blob/master/lab-11-5-mnist_cnn_ensemble_layers.py>

<https://github.com/hunkim/DeepLearningZeroToAll/blob/master/lab-11-X-mnist_cnn_low_memory.py>

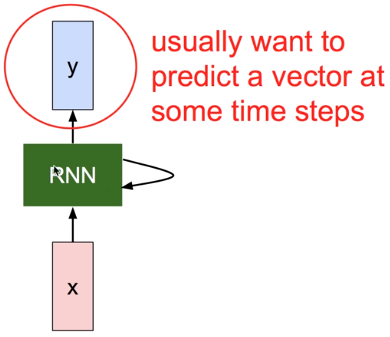
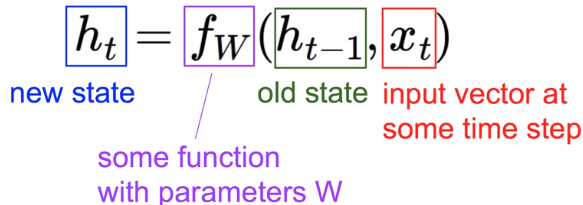
**ML\_Lec 12.**

**Sequence Data.**

****

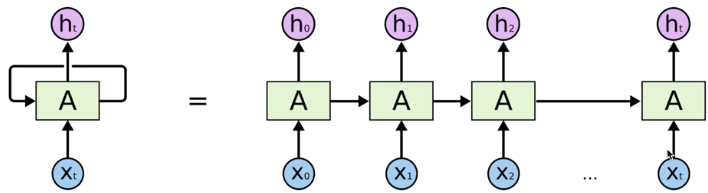
1. We don’t understand one word only.
2. We understand based on the previous words + this word. (time series)
3. NN/CNN cannot do this.
4. Notice: the same function and the same set of parameters are used at every time step.

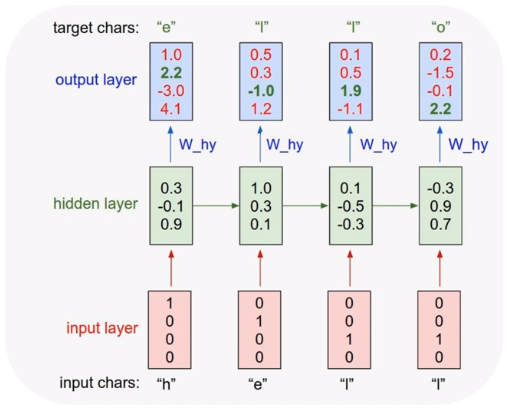
**Recurrent Neural Network. //순환신경망**



♠ Vanilla //The state consists of a single “hidden” vector h

**Character-level Language Model Example.**

****

****

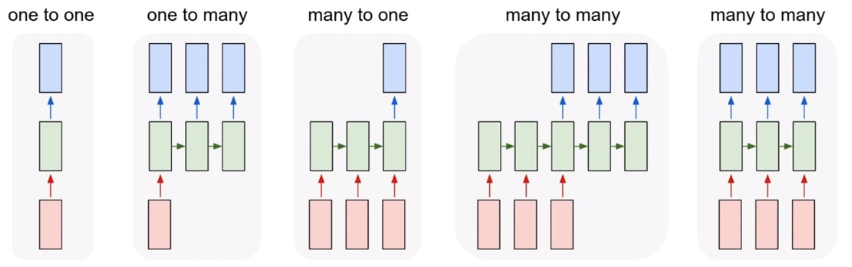
Error

Using Softmax

Vocabulary: [h, e, l, o]

Example Training Sequence: “hello”

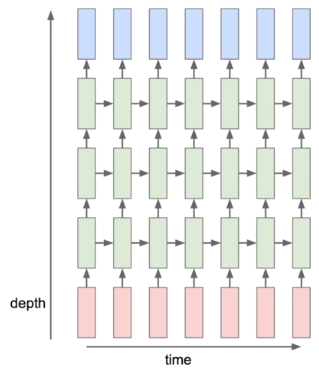
**RNN Applications.**



Vanilla Image Captioning Sentiment Classification Machine Translation Video Classification

1. Language Modeling
2. Speech Recognition
3. Machine Translation
4. Conversation Modeling/Question Answering
5. Image/Video Captioning
6. Image/Video/Dance Generation

**Multi-Layer RNN.**

****

**Training RNNs is Challenging.**

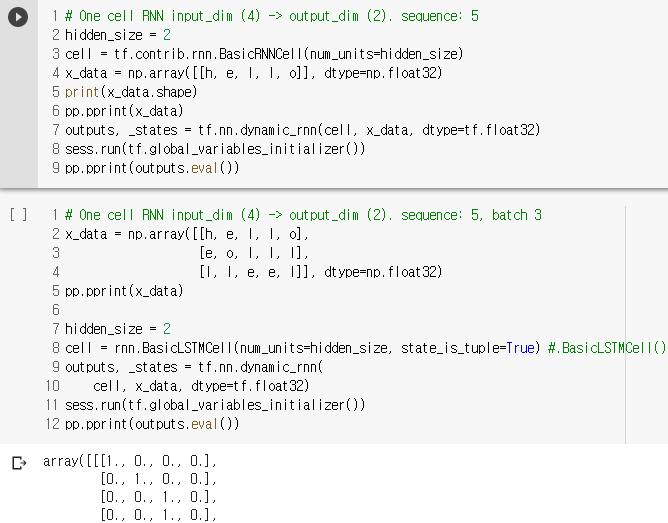
♠ Several Advanced Models

1. Long Short Term Memory (LSTM)
2. GRU by Cho et al.2014

**ML\_Lab 12-1.**

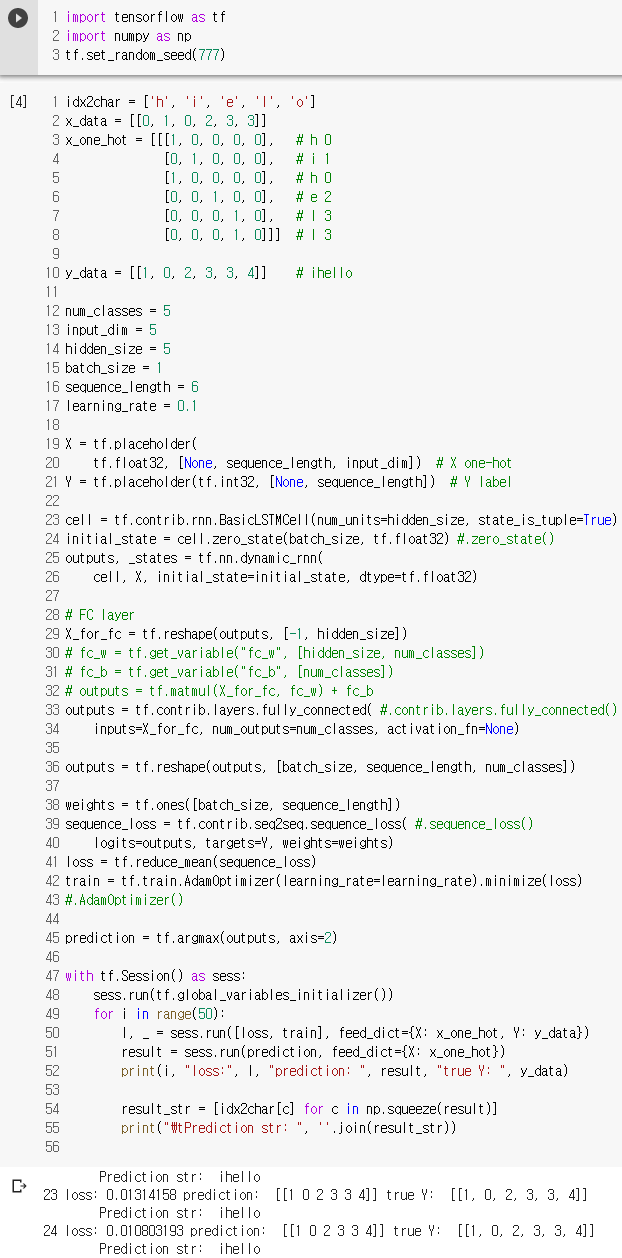
**Test.**





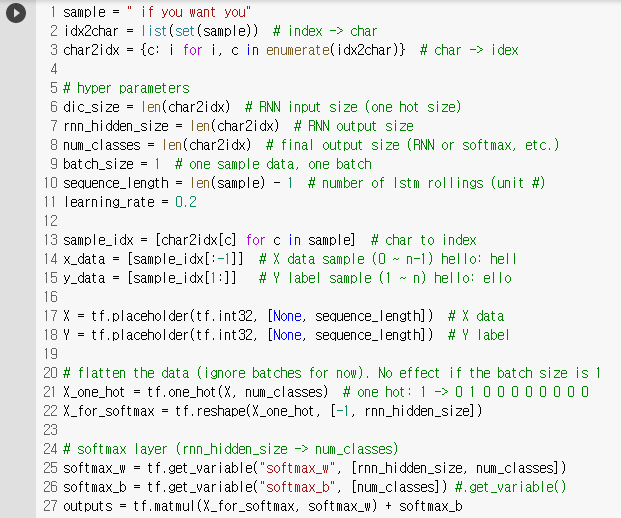
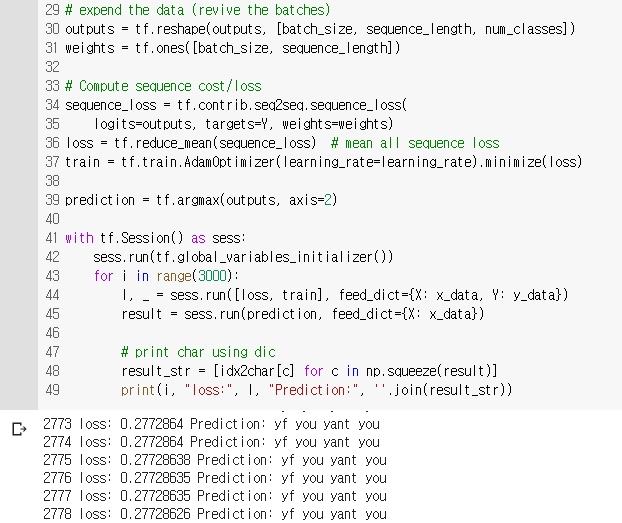
**ML\_Lab 12-2.**

**Test.**



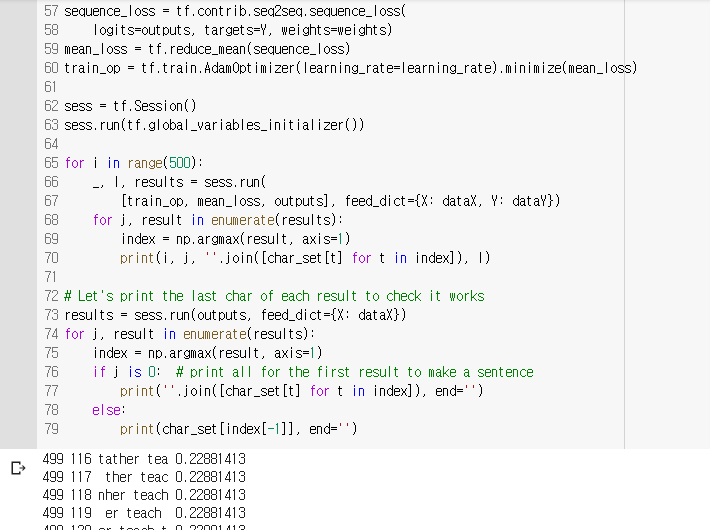
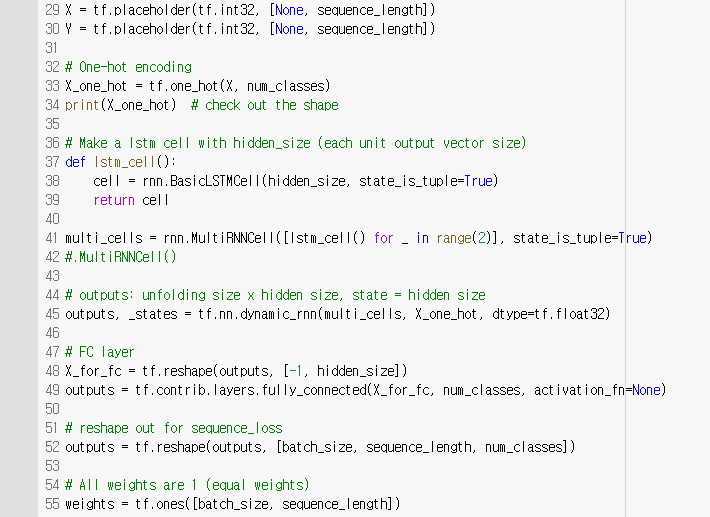
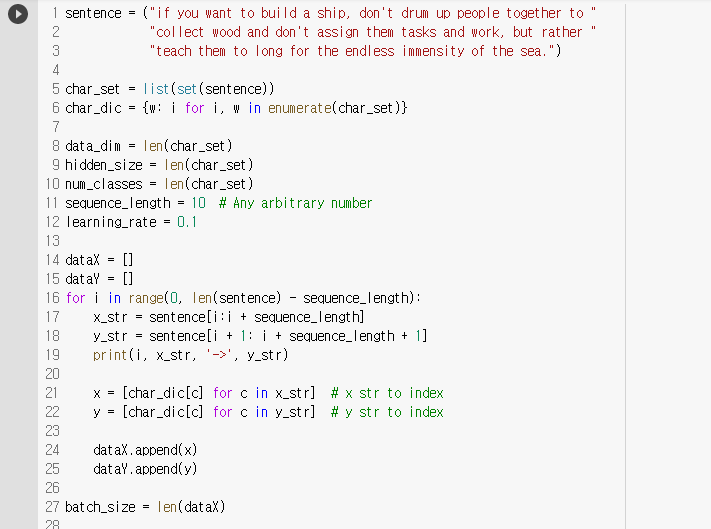
**ML\_Lab 12-3.**

**Test.**

**ML\_Lab 12-4.**

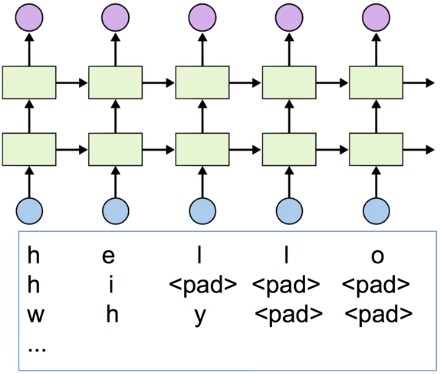
**Test.**



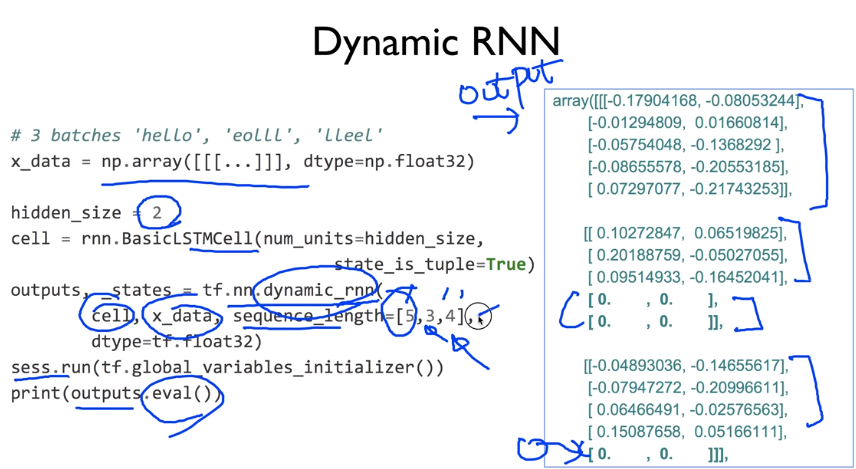
**ML\_Lab 12-5.**

**Different Sequence Length.**

Sequence\_length=[5, 2, 3]

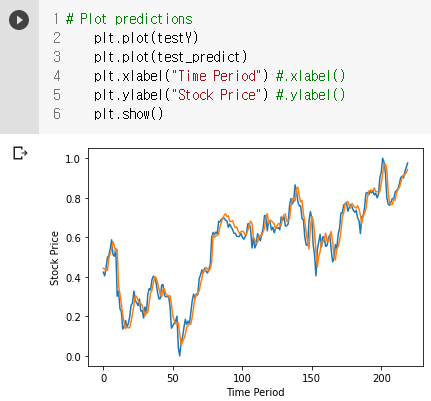
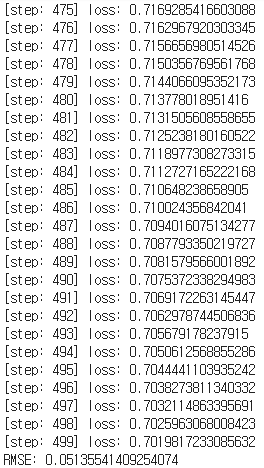
****

**Dynamic RNN.**

****

**ML\_Lab 12-6.**

**Test.**



<https://raw.githubusercontent.com/hunkim/DeepLearningZeroToAll/master/lab-12-5-rnn_stock_prediction.py>

**Memo.**

.constant() //상수

.Session() //tensor에 데이터를 넣어 흐르게 함.

.run() //실행

.add() //더하기

.placeholeder(), feed\_dict={a:a\_data} //변수, 값을 나중에 할당.

.Variable() //변수, 자동으로 업데이트.

.random\_normal(Shapes) //랜덤 값 반환

.reduce\_mean() //평균

.square() //제곱

.GradientDescentOptimizer() //미니 배치 확률적 경사하강법(SGD) 구현.

.minimize() //최소화

.global\_variables\_initializer() //.Variable()를 초기화.

.append() //append

.plot() //plot

.show() //show

.reduce\_sum() //총합

.assign() //.Variable()의 값 변경.

.compute\_gradients() //compute\_gradients

.apply\_gradients() //apply\_gradients

.matmul() //matmul

.loadtext() //text불러오기.

.set\_random\_seed() //랜덤 값 시드, 다른 환경에서도 같다.

.string\_input\_producer() //Queue, text를 Filename Queue에 쌓기.

.TextLineReader() //Queue, text를 Reader로 연결.

.read() //Queue, text읽기.

.decode\_csv() //Queue, text decode

.batch() //Queue, text batch

.Coordinator() //Queue, Coordinator생성.

.start\_queue\_runners() //Queue, Queue를 Thread로 시작.

.request\_stop() //Queue, 중지

.join() //Queue, 대기

.sigmoid() //S자 곡선

.log() //로그

.cast() //새로운 자료형

.equal() //값이 같은지

.softmax() //softmax

.arg\_max() //arg\_max

.one\_hot() //one\_hot

.reshape() //reshape

.softmax\_cross\_entropy\_with\_logits() //softmax\_cross\_entropy\_with\_logits

.format() //format

.flatten() //flatten

.PrettyPrinter() //PrettyPrinter

.InteractiveSession() //InteractiveSession

.array() //Array

.pprint() //Pprint

.shape() //Shape

.eval() //Eval

.squeeze() //Array 정리

.expand\_dims() //Array 정렬

.stack() //Array 쌓기

.ones\_like() //One으로 바꿈.

.zeros\_like() //Zero로 바꿈.

zip() //Zip

.nn.relu() //Relu

.random.randn() //지정 범위 내 랜덤 값 반환

.nn.dropout() //Dropout

.imshow() //Imshow

.nn.conv2d() //Conv2d

.swapaxes() //Swapaxes

.subplot() //Subplot

.nn.max\_pool() //Maxpool

.contrib.rnn.BasicRNNCell() //Basic RNN Cell

.nn.dynamic\_rnn() //Dynamic Rnn

.BasicLSTMCell() //Basic LSTM Cell

.zero\_state() //Zero State

.contrib.layers.fully\_connected() //Fully Connected

.ones() //Ones

.contrib.seq2seq.sequence\_loss() //Sequence Loss

.AdamOptimizer() //AdamOptimizer

.get\_variable() //Get Variable

.MultiRNNCell() //Multi RNN Cell

.min() //최소

.max() //최대

MinMaxScaler //MinMaxScaler

.xlabel() //X Label

.ylabel() //Y Label

//Lab으로 연결 https://colab.research.google.com/drive/1gaTpEufmhoK2CsEsNyfDDtyynQ\_HRpSu

//14폰트, 12폰트, 10폰트

//1. 1) a. \*♠

//0.71 1.34