To learn about the datasets we are using for neural n/w movie review analysis refer to

<https://keras.io/datasets/>

## IMDB Movie reviews sentiment classification

Dataset of 25,000 movies reviews from IMDB, labeled by sentiment (positive/negative). Reviews have been preprocessed, and each review is encoded as a [sequence](https://keras.io/preprocessing/sequence/) of word indexes (integers). For convenience, words are indexed by overall frequency in the dataset, so that for instance the integer "3" encodes the 3rd most frequent word in the data. This allows for quick filtering operations such as: "only consider the top 10,000 most common words, but eliminate the top 20 most common words".

As a convention, "0" does not stand for a specific word, but instead is used to encode any unknown word.

### Usage:

from keras.datasets import imdb

(x\_train, y\_train), (x\_test, y\_test) = imdb.load\_data(path="imdb.npz",

num\_words=None,

skip\_top=0,

maxlen=None,

seed=113,

start\_char=1,

oov\_char=2,

index\_from=3)

* **Returns:**
  + 2 tuples:
    - **x\_train, x\_test**: list of sequences, which are lists of indexes (integers). If the num\_words argument was specific, the maximum possible index value is num\_words-1. If the maxlen argument was specified, the largest possible sequence length is maxlen.
    - **y\_train, y\_test**: list of integer labels (1 or 0).
* **Arguments:**
  + **path**: if you do not have the data locally (at '~/.keras/datasets/' + path), it will be downloaded to this location.
  + **num\_words**: integer or None. Top most frequent words to consider. Any less frequent word will appear as oov\_char value in the sequence data.
  + **skip\_top**: integer. Top most frequent words to ignore (they will appear as oov\_char value in the sequence data).
  + **maxlen**: int. Maximum sequence length. Any longer sequence will be truncated.
  + **seed**: int. Seed for reproducible data shuffling.
  + **start\_char**: int. The start of a sequence will be marked with this character. Set to 1 because 0 is usually the padding character.
  + **oov\_char**: int. words that were cut out because of the num\_words or skip\_top limit will be replaced with this character.
  + **index\_from**: int. Index actual words with this index and higher.

ABOUT KERAS – keras.io

The core data structure of Keras is a **model**, a way to organize layers. The simplest type of model is the [Sequential](https://keras.io/getting-started/sequential-model-guide)model, a linear stack of layers.

Here is the Sequential model:

**from** keras.models **import** Sequential

model = Sequential()

Stacking layers is as easy as .add():

**from** keras.layers **import** Dense

model.add(Dense(units=64, activation='relu', input\_dim=100))

model.add(Dense(units=10, activation='softmax'))

Once your model looks good, configure its learning process with .compile():

model.compile(loss='categorical\_crossentropy',

optimizer='sgd',

metrics=['accuracy'])

TO REFER TO THE CODE REFER TO THE FOLLOWING LINK

<https://machinelearningmastery.com/predict-sentiment-movie-reviews-using-deep-learning/>