from pathlib import Path import string import re import joblib import json from collections import Counter import nltk from nltk.corpus import stopwords from nltk.stem import WordNetLemmatizer import nltk import random import pickle from time import time from nltk.stem.lancaster import LancasterStemmer from tensorflow.keras.utils import plot model from tensorflow.keras.models import Sequential from tensorflow.keras.layers import Embedding, Dense from tensorflow.keras.callbacks import ModelCheckpoint, EarlyStopping, ReduceLROnPlateau from tensorflow.python.keras.callbacks import TensorBoard tensorboard = TensorBoard(log dir=f"logs/{time()}") nltk.download('wordnet') [nltk data] Downloading package wordnet to C:\Users\Deepak [nltk data] Avudiappan\AppData\Roaming\nltk_data... [nltk data] Package wordnet is already up-to-date! Out[2]: True def load doc(jsonFile): with open(jsonFile, encoding="utf8") as file: Json data = json.loads(file.read()) return Json data In [4]: # data = load doc('intents.json') data = load doc('intents.json') # Data Preprocessing words, labels, $docs_x$, $docs_y = []$, [], []stemmer = LancasterStemmer() nltk.download('punkt') for intent in data['intents']: for pattern in intent['patterns']: wrds = nltk.word tokenize(pattern) words.extend(wrds) # instad of looping and appending we can extend the existing list docs x.append(wrds) docs_y.append(intent["tag"]) if intent['tag'] not in labels: labels.append(intent['tag']) words = [stemmer.stem(w.lower()) for w in words if w != '?'] # converting all words to lowercase words = sorted(list(set(words))) # removing all dublicate words labels = sorted(labels) # removing all dublicate labes # One Hot Encoding training, output = [], [] out_empty = [0 for _ in range(len(labels))] # Creating Bag of Words for x, doc in enumerate(docs_x): bag = []wrds = [stemmer.stem(w) for w in doc] for w in words: if w in wrds: bag.append(1) else: bag.append(0) output_row = out_empty[:] output row[labels.index(docs y[x])] = 1 training.append(bag) output.append(output row) [nltk data] Downloading package punkt to C:\Users\Deepak Avudiappan\AppData\Roaming\nltk data... [nltk data] Package punkt is already up-to-date! # converting to array training = np.array(training) output = np.array(output) training[0].shape Out[7]: (128,) output[0].shape Out[8]: (17,) #Predicting results with open("data.pickle", "wb") as f: pickle.dump((words, labels, training, output), f) early stopping = EarlyStopping(monitor='loss',patience=5) checkpoint = ModelCheckpoint("model-v1.h5", monitor="loss", mode="min", save best only = True, verbose=1) reduce lr = ReduceLROnPlateau(monitor = 'loss', factor = 0.2, patience = 3, verbose = 1, min delta = 0.0001) callbacks = [early stopping, checkpoint, reduce lr] model = Sequential() model.add(Dense(units=len(training[0]), activation = 'relu')) model.add(Dense(32, activation = 'relu')) model.add(Dense(16, activation = 'relu')) model.add(Dense(len(output[0]), activation = 'softmax')) model.compile(loss = 'categorical crossentropy', optimizer = 'adam', metrics=['accuracy']) history = model.fit(training, output,batch_size = 16, epochs=500, callbacks = [tensorboard,callbacks], verbo Epoch 1/500 1/11 [=>......] - ETA: 0s - loss: 2.8544 - accuracy: 0.0000e+00WARNING:tensorflow:Fro m C:\Users\Deepak Avudiappan\AppData\Roaming\Python\Python37\site-packages\tensorflow\python\ops\summary ops _v2.py:1277: stop (from tensorflow.python.eager.profiler) is deprecated and will be removed after 2020-07-0 $\overline{1}$. Instructions for updating: use `tf.profiler.experimental.stop` instead. 2/11 [====>.....] - ETA: 0s - loss: 2.8404 - accuracy: 0.0000e+00WARNING:tensorflow:Cal lbacks method `on train batch end` is slow compared to the batch time (batch time: 0.0070s vs `on train batc h end` time: 0.0790s). Check your callbacks. Epoch 00001: loss improved from inf to 2.83251, saving model to model-v1.h5 Epoch 2/500 1/11 [=>.....] - ETA: 0s - loss: 2.7879 - accuracy: 0.1250 Epoch 00002: loss improved from 2.83251 to 2.79474, saving model to model-v1.h5 Epoch 3/500 1/11 [=>.....] - ETA: 0s - loss: 2.7832 - accuracy: 0.1250 Epoch 00003: loss improved from 2.79474 to 2.75679, saving model to model-v1.h5 Epoch 4/500 1/11 [=>.....] - ETA: 0s - loss: 2.6684 - accuracy: 0.1875 Epoch 00004: loss improved from 2.75679 to 2.71085, saving model to model-v1.h5 Epoch 5/500 1/11 [=>.....] - ETA: 0s - loss: 2.6249 - accuracy: 0.2500 Epoch 00005: loss improved from 2.71085 to 2.64569, saving model to model-v1.h5 Epoch 6/500 1/11 [=>.....] - ETA: 0s - loss: 2.5218 - accuracy: 0.1875 Epoch 00006: loss improved from 2.64569 to 2.56844, saving model to model-v1.h5 Epoch 7/500 1/11 [=>.....] - ETA: 0s - loss: 2.4282 - accuracy: 0.4375 Epoch 00007: loss improved from 2.56844 to 2.47323, saving model to model-v1.h5 Epoch 8/500 1/11 [=>.....] - ETA: 0s - loss: 2.6204 - accuracy: 0.0625 Epoch 00008: loss improved from 2.47323 to 2.35994, saving model to model-v1.h5 Epoch 9/500 1/11 [=>.....] - ETA: 0s - loss: 2.5114 - accuracy: 0.1250 Epoch 00009: loss improved from 2.35994 to 2.24708, saving model to model-v1.h5 1/11 [=>.....] - ETA: 0s - loss: 2.0641 - accuracy: 0.5625 Epoch 00010: loss improved from 2.24708 to 2.11931, saving model to model-v1.h5 Epoch 11/500 1/11 [=>.....] - ETA: 0s - loss: 1.9171 - accuracy: 0.6250 Epoch 00011: loss improved from 2.11931 to 1.98575, saving model to model-v1.h5 Epoch 12/500 1/11 [=>.....] - ETA: 0s - loss: 1.7662 - accuracy: 0.6250 Epoch 00012: loss improved from 1.98575 to 1.84040, saving model to model-v1.h5 Epoch 13/500 1/11 [=>......] - ETA: 0s - loss: 1.7519 - accuracy: 0.5000 Epoch 00013: loss improved from 1.84040 to 1.69979, saving model to model-v1.h5 Epoch 14/500 1/11 [=>.....] - ETA: 0s - loss: 1.7061 - accuracy: 0.5000 Epoch 00014: loss improved from 1.69979 to 1.55000, saving model to model-v1.h5 Epoch 15/500 1/11 [=>.....] - ETA: 0s - loss: 1.1444 - accuracy: 0.7500 Epoch 00015: loss improved from 1.55000 to 1.40511, saving model to model-v1.h5 Epoch 16/500 1/11 [=>.....] - ETA: 0s - loss: 1.1718 - accuracy: 0.5625 Epoch 00016: loss improved from 1.40511 to 1.26352, saving model to model-v1.h5 Epoch 17/500 1/11 [=>.....] - ETA: 0s - loss: 1.1315 - accuracy: 0.6875 Epoch 00017: loss improved from 1.26352 to 1.12006, saving model to model-v1.h5 1/11 [=>.....] - ETA: 0s - loss: 0.7486 - accuracy: 1.0000 Epoch 00018: loss improved from 1.12006 to 0.98514, saving model to model-v1.h5 Epoch 19/500 1/11 [=>.....] - ETA: 0s - loss: 0.7401 - accuracy: 1.0000 Epoch 00019: loss improved from 0.98514 to 0.85112, saving model to model-v1.h5 Epoch 20/500 1/11 [=>.....] - ETA: 0s - loss: 0.6172 - accuracy: 0.8125 Epoch 00020: loss improved from 0.85112 to 0.74179, saving model to model-v1.h5 Epoch 21/500 1/11 [=>.....] - ETA: 0s - loss: 0.6334 - accuracy: 1.0000 Epoch 00021: loss improved from 0.74179 to 0.63722, saving model to model-v1.h5 Epoch 22/500 1/11 [=>.....] - ETA: 0s - loss: 0.4522 - accuracy: 1.0000 Epoch 00022: loss improved from 0.63722 to 0.54734, saving model to model-v1.h5 Epoch 23/500 1/11 [=>.....] - ETA: 0s - loss: 0.5302 - accuracy: 0.9375 Epoch 00023: loss improved from 0.54734 to 0.46896, saving model to model-v1.h5 Epoch 24/500 1/11 [=>.....] - ETA: 0s - loss: 0.4271 - accuracy: 0.9375 Epoch 00024: loss improved from 0.46896 to 0.40477, saving model to model-v1.h5 Epoch 25/500 1/11 [=>.....] - ETA: 0s - loss: 0.6125 - accuracy: 0.8750 Epoch 00025: loss improved from 0.40477 to 0.35029, saving model to model-v1.h5 Epoch 26/500 1/11 [=>.....] - ETA: 0s - loss: 0.2379 - accuracy: 1.0000 Epoch 00026: loss improved from 0.35029 to 0.30240, saving model to model-v1.h5 Epoch 27/500 1/11 [=>.....] - ETA: Os - loss: 0.2897 - accuracy: 0.9375 Epoch 00027: loss improved from 0.30240 to 0.26973, saving model to model-v1.h5 Epoch 28/500 1/11 [=>.....] - ETA: 0s - loss: 0.2023 - accuracy: 0.9375 Epoch 00028: loss improved from 0.26973 to 0.24190, saving model to model-v1.h5 Epoch 29/500 1/11 [=>.....] - ETA: Os - loss: 0.2170 - accuracy: 0.9375 Epoch 00029: loss improved from 0.24190 to 0.21946, saving model to model-v1.h5 Epoch 30/500 1/11 [=>.....] - ETA: 0s - loss: 0.1499 - accuracy: 1.0000 Epoch 00030: loss improved from 0.21946 to 0.20609, saving model to model-v1.h5 Epoch 31/500 1/11 [=>.....] - ETA: 0s - loss: 0.2459 - accuracy: 0.9375 Epoch 00031: loss improved from 0.20609 to 0.19112, saving model to model-v1.h5 Epoch 32/500 1/11 [=>.....] - ETA: 0s - loss: 0.0987 - accuracy: 1.0000 Epoch 00032: loss improved from 0.19112 to 0.17112, saving model to model-v1.h5 Epoch 33/500 1/11 [=>.....] - ETA: 0s - loss: 0.0897 - accuracy: 1.0000 Epoch 00033: loss improved from 0.17112 to 0.16493, saving model to model-v1.h5 Epoch 34/500 1/11 [=>.....] - ETA: 0s - loss: 0.1307 - accuracy: 1.0000 Epoch 00034: loss improved from 0.16493 to 0.14692, saving model to model-v1.h5 Epoch 35/500 1/11 [=>.....] - ETA: Os - loss: 0.0537 - accuracy: 1.0000 Epoch 00035: loss improved from 0.14692 to 0.14059, saving model to model-v1.h5 Epoch 36/500 1/11 [=>.....] - ETA: 0s - loss: 0.0908 - accuracy: 1.0000 Epoch 00036: loss did not improve from 0.14059 Epoch 37/500 1/11 [=>.....] - ETA: 0s - loss: 0.2301 - accuracy: 0.9375 Epoch 00037: loss did not improve from 0.14059 Epoch 38/500 1/11 [=>.....] - ETA: 0s - loss: 0.2670 - accuracy: 0.9375 Epoch 00038: loss improved from 0.14059 to 0.11977, saving model to model-v1.h5 Epoch 39/500 1/11 [=>.....] - ETA: 0s - loss: 0.0571 - accuracy: 1.0000 Epoch 00039: loss improved from 0.11977 to 0.11762, saving model to model-v1.h5 Epoch 40/500 1/11 [=>.....] - ETA: 0s - loss: 0.0397 - accuracy: 1.0000 Epoch 00040: loss improved from 0.11762 to 0.10949, saving model to model-v1.h5 Epoch 41/500 1/11 [=>.....] - ETA: 0s - loss: 0.0758 - accuracy: 1.0000 Epoch 00041: loss improved from 0.10949 to 0.10804, saving model to model-v1.h5 Epoch 42/500 1/11 [=>.....] - ETA: 0s - loss: 0.0365 - accuracy: 1.0000 Epoch 00042: loss improved from 0.10804 to 0.09941, saving model to model-v1.h5 Epoch 43/500 1/11 [=>.....] - ETA: 0s - loss: 0.0502 - accuracy: 1.0000 Epoch 00043: loss improved from 0.09941 to 0.09639, saving model to model-v1.h5 Epoch 44/500 1/11 [=>.....] - ETA: 0s - loss: 0.0816 - accuracy: 1.0000 Epoch 00044: loss improved from 0.09639 to 0.09387, saving model to model-v1.h5 Epoch 45/500 1/11 [=>.....] - ETA: 0s - loss: 0.0246 - accuracy: 1.0000 Epoch 00045: loss did not improve from 0.09387 Epoch 46/500 1/11 [=>.....] - ETA: 0s - loss: 0.1585 - accuracy: 0.9375 Epoch 00046: loss improved from 0.09387 to 0.08745, saving model to model-v1.h5 Epoch 47/500 1/11 [=>.....] - ETA: Os - loss: 0.0426 - accuracy: 1.0000 Epoch 00047: loss improved from 0.08745 to 0.08585, saving model to model-v1.h5 Epoch 48/500 1/11 [=>.....] - ETA: 0s - loss: 0.0290 - accuracy: 1.0000 Epoch 00048: loss improved from 0.08585 to 0.08260, saving model to model-v1.h5 Epoch 49/500 1/11 [=>.....] - ETA: 0s - loss: 0.1533 - accuracy: 0.9375 Epoch 00049: loss did not improve from 0.08260 Epoch 50/500 1/11 [=>.....] - ETA: 0s - loss: 0.0735 - accuracy: 0.9375 Epoch 00050: loss improved from 0.08260 to 0.08142, saving model to model-v1.h5 Epoch 51/500 1/11 [=>.....] - ETA: Os - loss: 0.0228 - accuracy: 1.0000 Epoch 00051: loss improved from 0.08142 to 0.07933, saving model to model-v1.h5 Epoch 52/500 1/11 [=>......] - ETA: 0s - loss: 0.1914 - accuracy: 0.9375 Epoch 00052: loss improved from 0.07933 to 0.07676, saving model to model-v1.h5 Epoch 53/500 1/11 [=>......] - ETA: 0s - loss: 0.1347 - accuracy: 0.9375 Epoch 00053: loss did not improve from 0.07676 Epoch 54/500 1/11 [=>.....] - ETA: 0s - loss: 0.0635 - accuracy: 0.9375 Epoch 00054: loss improved from 0.07676 to 0.07495, saving model to model-v1.h5 Epoch 55/500 1/11 [=>.....] - ETA: Os - loss: 0.0205 - accuracy: 1.0000 Epoch 00055: loss did not improve from 0.07495 Epoch 56/500 1/11 [=>.....] - ETA: 0s - loss: 0.0330 - accuracy: 1.0000 Epoch 00056: loss improved from 0.07495 to 0.07154, saving model to model-v1.h5 Epoch 57/500 1/11 [=>.....] - ETA: 0s - loss: 0.0510 - accuracy: 1.0000 Epoch 00057: loss did not improve from 0.07154 Epoch 58/500 1/11 [=>.....] - ETA: 0s - loss: 0.0620 - accuracy: 1.0000 Epoch 00058: loss did not improve from 0.07154 Epoch 59/500 1/11 [=>.....] - ETA: Os - loss: 0.0797 - accuracy: 1.0000 Epoch 00059: loss improved from 0.07154 to 0.06925, saving model to model-v1.h5 Epoch 60/500 1/11 [=>.....] - ETA: 0s - loss: 0.1396 - accuracy: 0.8750 Epoch 00060: loss did not improve from 0.06925 Epoch 61/500 1/11 [=>.....] - ETA: 0s - loss: 0.1574 - accuracy: 0.8750 Epoch 00061: loss did not improve from 0.06925 Epoch 62/500 1/11 [=>.....] - ETA: 0s - loss: 0.0542 - accuracy: 1.0000 Epoch 00062: loss did not improve from 0.06925 Epoch 00062: ReduceLROnPlateau reducing learning rate to 0.00020000000949949026. Epoch 63/500 1/11 [=>] - ETA: 0s - loss: 0.0612 - accuracy: 0.9375 Epoch 00063: loss improved from 0.06925 to 0.06449, saving model to model-v1.h5 Epoch 64/500 1/11 [=>.....] - ETA: 0s - loss: 0.1063 - accuracy: 0.9375 Epoch 00064: loss improved from 0.06449 to 0.06395, saving model to model-v1.h5 Epoch 65/500 1/11 [=>.....] - ETA: 0s - loss: 0.0136 - accuracy: 1.0000 Epoch 00065: loss improved from 0.06395 to 0.06374, saving model to model-v1.h5 Epoch 66/500 1/11 [=>.....] - ETA: 0s - loss: 0.0578 - accuracy: 0.9375 Epoch 00066: loss did not improve from 0.06374 Epoch 67/500 1/11 [=>.....] - ETA: Os - loss: 0.0822 - accuracy: 1.0000 Epoch 00067: loss improved from 0.06374 to 0.06318, saving model to model-v1.h5 Epoch 68/500 1/11 [=>.....] - ETA: 0s - loss: 0.1476 - accuracy: 0.9375 Epoch 00068: loss did not improve from 0.06318 Epoch 69/500 1/11 [=>.....] - ETA: 0s - loss: 0.0600 - accuracy: 0.9375 Epoch 00069: loss did not improve from 0.06318 1/11 [=>......] - ETA: 0s - loss: 0.0112 - accuracy: 1.0000 Epoch 00070: loss improved from 0.06318 to 0.06285, saving model to model-v1.h5 Epoch 71/500 1/11 [=>.....] - ETA: 0s - loss: 0.1657 - accuracy: 0.9375 Epoch 00071: loss did not improve from 0.06285 Epoch 72/500 1/11 [=>.....] - ETA: 0s - loss: 0.0096 - accuracy: 1.0000 Epoch 00072: loss improved from 0.06285 to 0.06238, saving model to model-v1.h5 Epoch 73/500 1/11 [=>......] - ETA: 0s - loss: 0.0725 - accuracy: 0.9375 Epoch 00073: loss improved from 0.06238 to 0.06219, saving model to model-v1.h5 Epoch 74/500 1/11 [=>.....] - ETA: 0s - loss: 0.0381 - accuracy: 1.0000 Epoch 00074: loss did not improve from 0.06219 Epoch 75/500 1/11 [=>.....] - ETA: 0s - loss: 0.0130 - accuracy: 1.0000 Epoch 00075: loss improved from 0.06219 to 0.06179, saving model to model-v1.h5 Epoch 76/500 1/11 [=>.....] - ETA: 0s - loss: 0.1122 - accuracy: 0.9375 Epoch 00076: loss did not improve from 0.06179 Epoch 77/500 1/11 [=>.....] - ETA: 0s - loss: 0.0107 - accuracy: 1.0000 Epoch 00077: loss did not improve from 0.06179 Epoch 78/500 1/11 [=>.....] - ETA: 0s - loss: 0.0679 - accuracy: 1.0000 Epoch 00078: loss did not improve from 0.06179 Epoch 00078: ReduceLROnPlateau reducing learning rate to 4.0000001899898055e-05. Epoch 79/500 1/11 [=>.....] - ETA: 0s - loss: 0.0568 - accuracy: 0.9375 Epoch 00079: loss improved from 0.06179 to 0.06112, saving model to model-v1.h5 Epoch 80/500 1/11 [=>.....] - ETA: 0s - loss: 0.0347 - accuracy: 1.0000 Epoch 00080: loss did not improve from 0.06112 Epoch 81/500 1/11 [=>.....] - ETA: 0s - loss: 0.1385 - accuracy: 0.9375 Epoch 00081: loss did not improve from 0.06112 Epoch 82/500 1/11 [=>.....] - ETA: 0s - loss: 0.0511 - accuracy: 1.0000 Epoch 00082: loss did not improve from 0.06112 Epoch 00082: ReduceLROnPlateau reducing learning rate to 8.000000525498762e-06. Epoch 83/500 1/11 [=>.....] - ETA: 0s - loss: 0.0481 - accuracy: 1.0000 Epoch 00083: loss improved from 0.06112 to 0.06097, saving model to model-v1.h5 Epoch 84/500 1/11 [=>......] - ETA: 0s - loss: 0.1289 - accuracy: 0.8750 Epoch 00084: loss did not improve from 0.06097 Epoch 85/500 1/11 [=>.....] - ETA: 0s - loss: 0.0060 - accuracy: 1.0000 Epoch 00085: loss improved from 0.06097 to 0.06095, saving model to model-v1.h5 Epoch 86/500 1/11 [=>.....] - ETA: 0s - loss: 0.1209 - accuracy: 0.8750 Epoch 00086: loss improved from 0.06095 to 0.06091, saving model to model-v1.h5 Epoch 00086: ReduceLROnPlateau reducing learning rate to 1.6000001778593287e-06. Epoch 87/500 1/11 [=>.....] - ETA: Os - loss: 0.0624 - accuracy: 1.0000 Epoch 00087: loss improved from 0.06091 to 0.06090, saving model to model-v1.h5 Epoch 88/500 1/11 [=>.....] - ETA: 0s - loss: 0.0076 - accuracy: 1.0000 Epoch 00088: loss improved from 0.06090 to 0.06089, saving model to model-v1.h5 Epoch 89/500 1/11 [=>.....] - ETA: 0s - loss: 0.0100 - accuracy: 1.0000 Epoch 00089: loss did not improve from 0.06089 Epoch 00089: ReduceLROnPlateau reducing learning rate to 3.200000264769187e-07. Epoch 90/500 1/11 [=>.....] - ETA: 0s - loss: 0.0623 - accuracy: 0.9375 Epoch 00090: loss improved from 0.06089 to 0.06088, saving model to model-v1.h5 Epoch 91/500 1/11 [=>.....] - ETA: 0s - loss: 0.1410 - accuracy: 0.8750 Epoch 00091: loss did not improve from 0.06088 Epoch 92/500 1/11 [=>.....] - ETA: 0s - loss: 0.0662 - accuracy: 0.9375 Epoch 00092: loss did not improve from 0.06088 Epoch 00092: ReduceLROnPlateau reducing learning rate to 6.400000529538374e-08. Epoch 93/500 1/11 [=>.....] - ETA: 0s - loss: 0.0770 - accuracy: 0.9375 Epoch 00093: loss improved from 0.06088 to 0.06088, saving model to model-v1.h5 Epoch 94/500 1/11 [=>......] - ETA: 0s - loss: 0.1029 - accuracy: 0.9375 Epoch 00094: loss did not improve from 0.06088 Epoch 95/500 1/11 [=>.....] - ETA: 0s - loss: 0.0931 - accuracy: 0.9375 Epoch 00095: loss did not improve from 0.06088 Epoch 00095: ReduceLROnPlateau reducing learning rate to 1.2800001059076749e-08. Epoch 96/500 1/11 [=>.....] - ETA: 0s - loss: 0.0095 - accuracy: 1.0000 Epoch 00096: loss improved from 0.06088 to 0.06088, saving model to model-v1.h5 Epoch 97/500 1/11 [=>.....] - ETA: 0s - loss: 0.0977 - accuracy: 0.9375 Epoch 00097: loss improved from 0.06088 to 0.06088, saving model to model-v1.h5 Epoch 98/500 1/11 [=>.....] - ETA: 0s - loss: 0.0948 - accuracy: 0.9375 Epoch 00098: loss did not improve from 0.06088 Epoch 00098: ReduceLROnPlateau reducing learning rate to 2.5600002118153498e-09. Epoch 99/500 1/11 [=>.....] - ETA: 0s - loss: 0.0433 - accuracy: 1.0000 Epoch 00099: loss improved from 0.06088 to 0.06088, saving model to model-v1.h5 - loss: 0.0609 - accuracy: 0.9691 11/11 0s 5ms/step Epoch 100/500 1/11 [=>.....] - ETA: 0s - loss: 0.1313 - accuracy: 0.9375 Epoch 00100: loss did not improve from 0.06088 Epoch 101/500 1/11 [=>......] - ETA: 0s - loss: 0.0611 - accuracy: 0.9375 Epoch 00101: loss did not improve from 0.06088 Epoch 00101: ReduceLROnPlateau reducing learning rate to 5.1200004236307e-10. Epoch 102/500 1/11 [=>.....] - ETA: 0s - loss: 0.0875 - accuracy: 1.0000 Epoch 00102: loss did not improve from 0.06088 Epoch 103/500 1/11 [=>.....] - ETA: Os - loss: 0.0710 - accuracy: 0.9375 Epoch 00103: loss did not improve from 0.06088 Epoch 104/500 1/11 [=>.....] - ETA: 0s - loss: 0.0655 - accuracy: 1.0000 Epoch 00104: loss did not improve from 0.06088 Epoch 00104: ReduceLROnPlateau reducing learning rate to 1.0240001069306004e-10. with open("data.pickle", "rb") as f: words, labels, training, output = pickle.load(f) In [14]: def bag_of_words(s, words): bag = [0 for _ in range(len(words))] s words = nltk.word tokenize(s) s_words = [stemmer.stem(word.lower()) for word in s_words] for se in s words: for i, w in enumerate(words): **if** w == se: bag[i] = 1return np.array(bag) def chat(): print("start talking with the bot (quit to exit)") inp = input("You: ").lower() results = np.array([bag of words(inp, words)]) results = model.predict(results)[0] result index = np.argmax(results) #returns index of the greatest value in our list tag = labels[result_index] if inp in ['bye','break', 'exit', 'quit', 'bye', 'close']: for tg in data["intents"]: if tg["tag"] == tag: responces = tg["responses"] print("Bot: ", random.choice(responces)) elif results[result index] > 0.70: #W setting the p-value for tg in data["intents"]: if tg["tag"] == tag: responces = tg["responses"] print("Bot: ", random.choice(responces)) print("Bot: Sorry I did't get that, As I'm limited to the purpose!") chat() start talking with the bot (quit to exit) Bot: Hello there, How may I help you? Bot: I'm good Bot: Hi, I'm Fiana Bot: I am Fiana 🕍 🕻 , made by Deepak Avudiappan on 27-2-2021. I predict whether a tumor is benign or maligna nt using the power of Artificial Neural Network of 85.15% accuracy. Note that I'm not accurate all the time I have a 14.85% chance of making mistake!, share a photo of the affected area with me I'll check Bot: Welcome Bot: I can recognize benign and malignant tumor by analyzing an image of the affected area. Bot: A tumor is an abnormal growth of body tissue. Tumors can be cancerous (malignant) or noncancerous (ben ign). Bot: Sure 😂, When the cells in the tumor are normal, it is benign. Bot: Sure 😂, When cells are abnormal and grows uncontrollably, they are cancerous cells, and the tumor is malignant. Bot: Welcome bag of words('hii', words) v = np.array([bag_of_words("hello", words)]) #y pred.reshape(len(y_pred),1) # model.predict(v) v[0].shape Out[19]: (128,) a = model.predict(v)[0] Out[21]: array([9.2154266e-08, 1.5848030e-04, 2.6878633e-03, 9.9602973e-01, 3.6097475e-07, 7.5720295e-06, 1.1555809e-06, 5.0292651e-06, 8.4550997e-08, 1.0590693e-08, 4.6956367e-04, 5.4820634e-07, 3.3090203e-07, 1.1169032e-04, 2.3647017e-05, 5.0249603e-04, 1.3319440e-06], dtype=float32) np.argmax(a) Out[22]: 3 labels Out[23]: ['benign', 'detect', 'end', 'greet', 'happy', 'malignant', 'menu', 'name', 'normal', 'options', 'sad', 'safeBM', 'search', 'tumor', 'upload', 'username', 'wish'] In [24]: model.summary() Model: "sequential" Output Shape Layer (type) Param # dense (Dense) (None, 128) 16512 dense_1 (Dense) (None, 32) 4128 dense 2 (Dense) (None, 16) 528 dense 3 (Dense) 289 (None, 17) Total params: 21,457 Trainable params: 21,457 Non-trainable params: 0 # !mkdir -p saved model model.save('saved model/my model') WARNING:tensorflow:From C:\Users\Deepak Avudiappan\AppData\Roaming\Python\Python37\site-packages\tensorflow \python\training\tracking\tracking.py:111: Model.state_updates (from tensorflow.python.keras.engine.trainin g) is deprecated and will be removed in a future version. Instructions for updating: This property should not be used in TensorFlow 2.0, as updates are applied automatically. WARNING:tensorflow:From C:\Users\Deepak Avudiappan\AppData\Roaming\Python\Python37\site-packages\tensorflow \python\training\tracking\tracking.py:111: Layer.updates (from tensorflow.python.keras.engine.base layer) is deprecated and will be removed in a future version. Instructions for updating: This property should not be used in TensorFlow 2.0, as updates are applied automatically. INFO:tensorflow:Assets written to: saved model/my model\assets import tensorflow as tf new model = tf.keras.models.load model('saved model/my model') v1 = np.array([bag of words("hello", words)]) new model.predict(v1) Out[28]: array([[9.2154266e-08, 1.5848030e-04, 2.6878633e-03, 9.9602973e-01, 3.6097475e-07, 7.5720295e-06, 1.1555809e-06, 5.0292651e-06, 8.4550997e-08, 1.0590693e-08, 4.6956367e-04, 5.4820634e-07, 3.3090203e-07, 1.1169032e-04, 2.3647017e-05, 5.0249603e-04, 1.3319440e-06]], dtype=float32) model.summary() Model: "sequential" Output Shape Param # Layer (type) 16512 dense (Dense) (None, 128) 4128 dense 1 (Dense) (None, 32) dense 2 (Dense) (None, 16) 528 dense 3 (Dense) (None, 17) Total params: 21,457 Trainable params: 21,457 Non-trainable params: 0 #CLI> tensorboard --logdir=logs/ data is in logs folder

import pandas as pd
import numpy as np

import matplotlib.pyplot as plt

