

Increasing the amount of layers will lead to higher accuracy in the chatbot. When we increase the epochs to 20 the variables change, the loss goes a little down but the accuracy stays about the same.

```

30 classes
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:62: VisibleDeprecationWarning: Creating an ndarray from ragged nested sequences (which is a list-or-tuple of lists-or-tuples of ndarrays) may degrade the array's dtype: <class 'numpy.ndarray'>
Epoch 1/20
29/29 [=====] - 1s 11ms/step - loss: 3.2245 - accuracy: 0.1367 - val_loss: 3.6677 - val_accuracy: 0.0000e+00
Epoch 2/20
29/29 [=====] - 0s 6ms/step - loss: 2.7968 - accuracy: 0.1722 - val_loss: 4.5173 - val_accuracy: 0.0000e+00
Epoch 3/20
29/29 [=====] - 0s 7ms/step - loss: 2.6579 - accuracy: 0.1667 - val_loss: 5.0645 - val_accuracy: 0.0000e+00
Epoch 4/20
29/29 [=====] - 0s 6ms/step - loss: 2.6440 - accuracy: 0.1722 - val_loss: 5.2388 - val_accuracy: 0.0000e+00
Epoch 5/20
29/29 [=====] - 0s 6ms/step - loss: 2.6416 - accuracy: 0.1722 - val_loss: 5.5037 - val_accuracy: 0.0000e+00
Epoch 6/20
29/29 [=====] - 0s 7ms/step - loss: 2.6342 - accuracy: 0.1722 - val_loss: 5.5914 - val_accuracy: 0.0000e+00
Epoch 7/20
29/29 [=====] - 0s 7ms/step - loss: 2.6371 - accuracy: 0.1722 - val_loss: 5.7367 - val_accuracy: 0.0000e+00
Epoch 8/20
29/29 [=====] - 0s 7ms/step - loss: 2.6340 - accuracy: 0.1722 - val_loss: 5.8353 - val_accuracy: 0.0000e+00
Epoch 9/20
29/29 [=====] - 0s 8ms/step - loss: 2.6331 - accuracy: 0.1722 - val_loss: 5.9897 - val_accuracy: 0.0000e+00
Epoch 10/20
29/29 [=====] - 0s 7ms/step - loss: 2.6306 - accuracy: 0.1722 - val_loss: 6.1377 - val_accuracy: 0.0000e+00
Epoch 11/20
29/29 [=====] - 0s 7ms/step - loss: 2.6313 - accuracy: 0.1722 - val_loss: 6.1986 - val_accuracy: 0.0000e+00
Epoch 12/20
29/29 [=====] - 0s 7ms/step - loss: 2.6307 - accuracy: 0.1722 - val_loss: 6.2563 - val_accuracy: 0.0000e+00
Epoch 13/20
29/29 [=====] - 0s 7ms/step - loss: 2.6288 - accuracy: 0.1689 - val_loss: 6.3382 - val_accuracy: 0.0000e+00
Epoch 14/20
29/29 [=====] - 0s 6ms/step - loss: 2.6303 - accuracy: 0.1722 - val_loss: 6.4586 - val_accuracy: 0.0000e+00
Epoch 15/20
29/29 [=====] - 0s 6ms/step - loss: 2.6299 - accuracy: 0.1722 - val_loss: 6.5066 - val_accuracy: 0.0000e+00
Epoch 16/20
29/29 [=====] - 0s 6ms/step - loss: 2.6297 - accuracy: 0.1722 - val_loss: 6.5510 - val_accuracy: 0.0000e+00
Epoch 17/20
29/29 [=====] - 0s 6ms/step - loss: 2.6290 - accuracy: 0.1722 - val_loss: 6.6335 - val_accuracy: 0.0000e+00
Epoch 18/20
29/29 [=====] - 0s 6ms/step - loss: 2.6310 - accuracy: 0.1722 - val_loss: 6.6460 - val_accuracy: 0.0000e+00
Epoch 19/20
29/29 [=====] - 0s 7ms/step - loss: 2.6291 - accuracy: 0.1567 - val_loss: 6.7136 - val_accuracy: 0.0000e+00
Epoch 20/20
29/29 [=====] - 0s 7ms/step - loss: 2.6284 - accuracy: 0.1722 - val_loss: 6.7627 - val_accuracy: 0.0000e+00
Finished training
ready
WARNING:tensorflow:5 out of the last 7 calls to <function Model.make_predict_function.<locals>.predict_function at 0x7f27df0dba70> triggered tf.function retracing. Tracing /usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:91: VisibleDeprecationWarning: Creating an ndarray from ragged nested sequences (which is a list-or-tuple of lists-or-tuples of ndarrays) may degrade the array's dtype: <class 'numpy.ndarray'>
Chatbot:No, I won't go with you

```

model.add(Dense(512, input_shape=(max_words,))) – The numbers of layers you define for the model, means 512 neurons. We changed this to 900, but then our system crashed so we were not able to understand this quite.

batch_size = 32 samples will be used to estimate mistake grade.

```

30 classes
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:62: VisibleDeprecationWarning: Creating an ndarray from ragged nested sequences (which is a list-or-tuple of lists-or-tuples of ndarrays) may degrade the array's dtype: <class 'numpy.ndarray'>
Epoch 1/4
29/29 [=====] - 1s 8ms/step - loss: 3.3930 - accuracy: 0.0878 - val_loss: 3.4098 - val_accuracy: 0.0000e+00
Epoch 2/4
29/29 [=====] - 0s 3ms/step - loss: 3.3648 - accuracy: 0.1222 - val_loss: 3.4215 - val_accuracy: 0.0000e+00
Epoch 3/4
29/29 [=====] - 0s 3ms/step - loss: 3.3401 - accuracy: 0.1089 - val_loss: 3.4355 - val_accuracy: 0.0000e+00
Epoch 4/4
29/29 [=====] - 0s 3ms/step - loss: 3.3123 - accuracy: 0.1089 - val_loss: 3.4545 - val_accuracy: 0.0000e+00
Finished training
ready
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:91: VisibleDeprecationWarning: Creating an ndarray from ragged nested sequences (which is a list-or-tuple of lists-or-tuples of ndarrays) may degrade the array's dtype: <class 'numpy.ndarray'>
Chatbot:I'm kidding. You know how sometimes you just become this "persona"? And you don't know how to quit?
Human:hello
Chatbot:the practically proposed when he found out we had the same dermatologist. I mean. Dr. Bonchowski is great an all, but he's not exactly relevant
Human:what are you
-----
ValueError                                Traceback (most recent call last)
<ipython-input-9-6ee25bed9514> in <module>()
    104 s = " "
    105 while s:
--> 106     category = getCategory(s)
    107     text = getRandomTextFromIndex(category)
    108     print("Chatbot:" + text)

~/local/lib/python3.7/dist-packages/keras_preprocessing/text.py in sequences_to_matrix(self, sequences, mode)
    413     x = np.zeros((len(sequences), num_words))
    414     for i, seq in enumerate(sequences):
--> 415         if not seq:
    416             continue
    417         counts = defaultdict(int)

ValueError: The truth value of an array with more than one element is ambiguous. Use a.any() or a.all()

```

Conclusion

We see the main problem with the complicated code and size of the text file being so big that our system crashed several times. Thus, if we had a little more time we

would be able to run more combinations for different variables and probably understand the outcome better.

Reference

Deepai.com. URL: <https://deepai.org/machine-learning-glossary-and-terms/epoch> accessed october 21, 2021.