AI & Automation Exam 2

05/06/2022

Mads Hegewald (EAAMHEG)

# Exercise 1

The first example of wordcloud is where I have used a file called “example.txt” in the directory “exercise1”, which is the transcript of the speech by Leonardo DiCaprio at the 2014 UN Climate Summit. First I read the txt file where I used .split() to get each individual word, followed by removing the comma “,” or dot “.” at the end of some of the words. After I added the words to a dictionary to keep track of the frequency of the words, but I did not want to add words such as “the”, “as”, “for” etc. So, I filtered these words out by using checking if the word was in the STOPWORDS from wordcloud. This is all done using the following code:

Text

Description automatically generated

Running the code shows the following wordcloud. One thing I noticed was how I wanted it to say “climate change” in one word but after some experimenting, I could not find a solid solution to that problem and stuck with splitting the words when a space between them occurred.

Text

Description automatically generated

For my second example I tried using a webtext from nltk.corpus called “pirate.txt” located at the same place as the example with “firefox.txt”. It seems like the text file contains a script of some sorts of the Pirates of the Caribbean movie. First we load in the words from webtext and use the nltk.FreqDist to create a dictionary with the frequencies of each word, this still contains all the stopwords mentioned previously. To sort these out I used a line from the example we had during one of the lessons, where we basically sort out all the words that are smaller than 3. This is done in the following code:

Text

Description automatically generated

After this we can simply just create a wordcloud object, where we generate the words from frequencies in the list “filter\_words” and we get the following wordcloud.

Text

Description automatically generated

# Exercise 2

## a)

* Neural net design
  + hidden layers
  + input and output
  + data input (what the net is looking at to learn how to play pong)
  + extra – compare different solutions

## b)

## c)

# Exercise 3

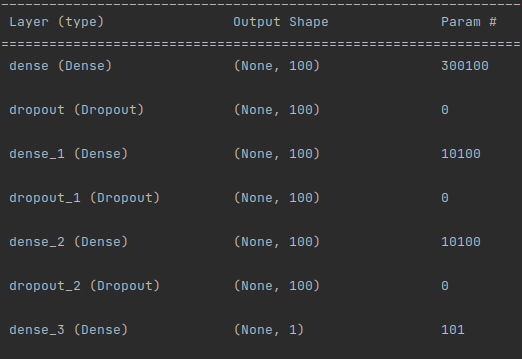
## a)

The initial accuracy of the model when run is around 9.3 at epoch 10:

For my first experiment I tried increasing the epochs to make it train for twice the amount (increased from 10 to 20), where I was expecting an increase in accuracy:

A slight increase of about 0.3, which is a nice increase but not a lot considering it trained for twice as long.

In my next experiment I tried adding another dense layer with 100 neurons followed by a dropout layer. So the model looks like this:

This did not increase the accuracy and actually worsened it, so just increasing layers this way did not produce a positive result.

For some of my final experiments I tried changing the overall design of the neural net, this includes the layers, activation functions, optimizer and loss function.

Overall I would suggest… to improve the neural network...

## b)

Word2Vec / CountVectorizer

Cleaning the data

Stemming or lemmatize

BERT

## c)

If the given score is closer to 0 it means a negative review and the closer to 1 the more positive the review, ??where 0.5 would be neutral.??  
  
In my own examples I wanted to try some sentences which might seem natural or are harder to define as positive or negative. Although the words to describe the plot, music, acting etc. might seem negative they can don't have to be in most circumstances (in my opinion). I did this in the first 2 examples, where I got the following scores:  
I thought the score for the first experiment would be more neutral (closer to 0.5), but the neural net predicted that it was a more negative review.

The second experiment result was closer to what I was expecting from the experiment since I was going for a neutral score.

Next I wanted to try the example we have been talking about during the lessons a lot of the time. Here I wrote some really positive sentences to start with and ended the review with a "NOT" indicating it was not a positive review.  
  
This result was exactly as I was expecting as well as what we have commented on during class, the neural net did not understand it and deemed it as an overly positive review.  
Lastly I wanted to try with mixed sentences, both negative and positive sentences in the review. Result is as follows:

Apparently the negative sentences outweighed the positive sentences by a lot, causing the neural net to deem the review as negative.

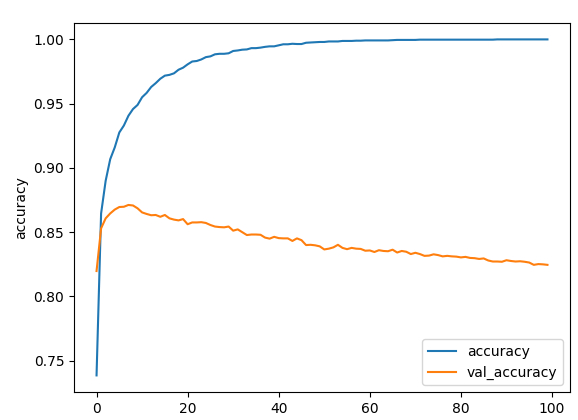
# Exercise 4

## a)

The initial run received following accuracy:



Running the simple model for 100 epochs yields:

We can see that it quickly reaches an accuracy greater than 0.95 in 20 epochs, where it slowly increases after. The increase in accuracy gets smaller and the loss becomes smaller as well the closer it gets to an accuracy of 1, but once it reaches around 90 epochs it reaches 1 in accuracy.

For the other small adjustments I made...

## b)

The first review I gave the model was a negative review on the movie “Morbius” where it only had a rating of 1/10 stars. The model defined the review as a more negative, which is correct in this case. I would have predicted the score to in the range of 0 to 0.2 since the rating was bad.



The second review I gave the model was a positive review of the movie “Mission Impossible Fallout” with a rating of 9/10. Here I expected a result close to 1 since the review was overly positive, but it only scored it at around 0.75.



## c)

Since we have added a hidden layer to the neural net…

explain val\_accuracy

## d)

If we can see that the model perform really well on the training data, but when it comes to actual samples it perform not so well, it means that the model is overfitting. There are a few ways to handle overfitting, one is changing the complexity of the neural net. If the neural net is too complex it could lead to overfitting. Another way for delaing with overfitting is regularization. In this case I would recommend regularizing with either a dropout layer, batch normalization or L2 weight regularization.

The hyper-parameters we could change are epochs,

<https://machinelearningmastery.com/introduction-to-regularization-to-reduce-overfitting-and-improve-generalization-error/>

exercise is basically just finding hyperparameters **include plots**

## e)

# Exercise 5

## a)

To make it easier for myself too see which the most frequently used words were, i sorted the list of word\_frequencies by the frequency value.



We have "Denmark" which is used 20 times, next is "The" with 7 usages. The words "Norway" and "century" both occur 5 times, and then we have a lot of words sharing a frequency of 4.

## b)

To find out the number of nouns and verbs in the text, I looped through the words and checked the .pos\_ field on every word. This will return what type the word is, where "NOUN" for noun and "VERB" for verb. After this it's simply just adding them to their own list. We can then see all the nouns and verbs as well as getting the total amount of each, which is 126 nouns and 53 verbs. The following code represents what is described:





## c)

The summarize function summarizes the input text based on a given ratio which in this case is 0.2. This ratio determines the number of sentences of the original text to be chosen for the summary. In the original text we have 26 sentences and with a ratio of 0.2 we end up with 4 sentences. With a ratio of 0.1 we only get 2 sentences.  
The way it chooses the sentences is based on a variation of the TextRank algorithm.  


To get the 5 most frequent word I created a dictionary and added the words to it if they were not in the stopwords or were a symbol such as a parenthesis.  
Then I sorted the dictionary based on the value (frequency) and printed the top 5.  
The 5 most frequent words used with a ratio of 0.1 is as seen below:

## d)

Numbers in the text, included in what sentences????  
missing coding part

The numbers in the text include primarily dates and some land area measured in km2.  
To find the information that best describes Denmark we could look at the numbers and then the words in that sentence. If the sentence contains the word Denmark or DK we can say that it is more likely to better describe it. The flaw with this is that if it doesn't contain Denmark or DK it would be deemed as not important.