CS72/LING48 Computational Linguistics
Dartmouth College. Spring 2023

Exercises Week 5
Neural Networks

Structure of this week's exercises

This week you will study some code and answer questions about it. Please run these in Google Colab, to ensure that you have the appropriate packages and training conditions. (You can run it in your computer, but we make no promises regarding whether it can run or not).

Please work with two of these and turn in your report in a PDF/LibreOffice/Word file with screenshots of your work. Please turn this in before 11:59 pm of Sunday, May 7th, 2023.

Part 1: Cook Islands Māori Part of Speech Tagging

The file `cim-3pos.csv` has information for words in the Cook Islands Māori language. The lines have the following structure:

["-" , "kua", "qaere", "TAM"]

This is a list with four elements. Imagine you have a sentence like *Kua 'aere au* "I went". This list captures the information of the word "kua". The first three elements of the list are (i) the word that precedes "kua", which in this case is nothing, (ii) the word "kua" itself (the perfect aspect marker) and (iii) the word that follow "kua", in this case "qaere" *to go*. The last element of the list is the correct part of speech (POS) tag for kua, `TAM` "Tense aspect marker". In the CSV file for this week you'll find information for three types of parts of speech: {noun, preposition, verb}.

You will use this data to train a neural network. The file `hw5_1_nn_pos_cim.ipynb` has a basic neural network: (i) An input layer that has 1497 input neurons (the length of the one-hot encoded vector for this dataset) and calculates outputs to 12 neurons, with a ReLU activation function. (ii) A second, hidden layer that has 12 input neurons and calculates outputs for 6 neurons, with a ReLU activation function. (iii) A third, hidden layer, with 6 input neurons and a calculation onto one output neuron, with a ReLU activation function.

You need to do report the following in a PDF/LibreOffice/Word document:

(a) Run the program three times and record the training and test accuracy for each of the three runs. What is the average training accuracy? What is the average accuracy for the test set? How are the F1-scores behaving? (You can see more information how the predictions are working by looking at the predictions for the first ten items).

(b) Change the program so that the hidden layers have 48 and 24 neurons, and the output has 3 neurons. Run the modified program three times. What is the average training/test accuracy? How are the F1-scores behaving? (You can see more information how the predictions are working by looking at the predictions for the first ten items).

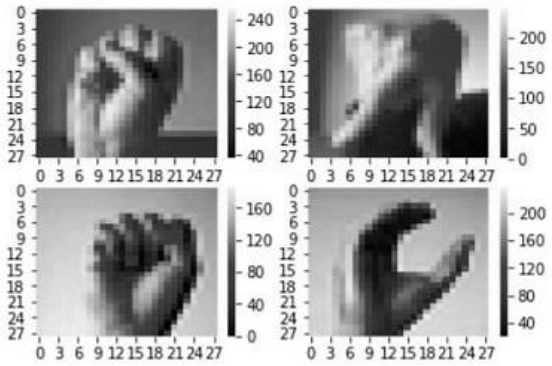
(c) Use the same settings as you did in step 'B' above and make one more change: Change the program so that it runs for 200 epochs. Run this three times. What is the average training and test accuracy? How are the F1-scores behaving? (You can see more information how the predictions are working by looking at the predictions for the first ten items).

(d) Please include screenshots to justify your answers.

Part 2: ConvNets and ASL Finger Spelling

A convolutional neural network (ConvNet/CNN) is optimized to understand visual data. One example is the program `hw5-2-cnn-asl.py`, which uses a ConvNet to learn 6 signs from ASL, the finger spelling letters A-F:

Each of the pictures is represented as the black-and-white pixel values for a 28x28 image. The image below has an example of what these values look like. If a pixel has a value of 255, it is white. If a pixel has a value of 0, it is black. All the values in the middle represent different saturations of grey.



The training set `sign-train-a-f.csv` has information for approximately 1100 different pictures for each sign. The information is presented as the black-and-white pixel values. The training set also contains the gold labels for each picture (a=0, b=1, c=2, d=3, e=4, f=5). The testing set `sign-test-a-f.csv` has information for 2063 pictures (331 'a', 432 'b', 310 'c', 245 'd', 498 'e' and 247 'f'). It uses the same format as the training set. Open the files in your Notepad++/BBEdit to take a look at what these files look like.

There are many good websites where you can learn the intuitions behind convolutional networks. These are some examples:

https://www.cs.ryerson.ca/~aharley/vis/conv/
https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/
https://www.freecodecamp.org/news/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050/
https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/
https://www.youtube.com/watch?v=iaSUYvmCekI (MIT Deep Learning: Convolutional Networks)

You need to write answers to these three questions:

(1) Study the links above and explain the code in the section *Convolutional Neural Network Structure* below. What is a kernel? What is pooling? Explain all of these as simply and plainly as you can.

(2) Run the program. Right now it's set to perform one epoch of training. How is the network behaving after one epoch of training? (Report this based on the accuracy, the precision and the recall for each of the letters).

(3) Change the program so that it runs for five epochs. How is the network behaving after five epochs of training? How have the values of accuracy, precision and recall changed for the ASL letters?

(4) Please remember to include screenshots for the questions #2 and #3.

(Keep in mind that, for this exercise, we will only look at hand shapes. However, in order to recognize other ASL signs, we would also need information from the person's face and from their use of the space around their head and torso)

Grading

You need to choose two of the topics. Each of the exercises is worth 2.5% Each will be graded according to the following qualitative scale:

"Completely clear" 100% of the score (2.5%): All of the questions are answered in a concise, clear and straightforward way. Relevant screenshots and examples are provided.

"Mostly clear" 75% of their score (1.875%): Most of the questions are answered in a concise, clear and straightforward manner. All of the questions are answered, but their explanation could be easier to understand or use more examples. The screenshots are present, but they are not directly relevant to the question.

"Somewhat clear" 50% of their score (1.25%): The questions are answered in a cursory way, or without delving into the concepts. Few or no examples are present. Screenshots are irrelevant.

"Unclear" 25% of their score (0.625%): The explanations show misunderstandings of the concepts presented. Many of the questions are not answered. No screenshots.

"No answer" 0% of their score. The student wrote no answer.