Practical 1: Chatbot – Simple ANN & Tranformers

Task 1.1

Create a simple chatbot that analyzes text responses typed by a user using an artificial neural network (ANN).

The minimum requirement is that the bot prompts a response from the user (with various possible prompts). When the user has typed the answer, the bot should analyze the text (using an ANN) and formulate a response based on the analysis.

A small dataset of product reviews that have been labelled as negative (0) or positive (1) is provided in the Files - Exercises - Lab 1 folder, along with some code needed to extract information.

A suggested approach is first to try to train a network on the given data.

When that task has been concluded, the model can be improved by finding more data, using a dataset with a broader range of labels, using word embeddings to create unique sentence embeddings, making the bot capable of extended dialogue or any other extension you want to pursue.

1 Task 1.2 Transformers Implementation

For this task, you will implement your transformer in PyTorch. You are instructed to follow this link: Tranformers in Pytorch.

2 Task 1.2 (Alternative)

If you find any problems with the previous code, links, or set-up (due to Amazon or anything else), we offer you another alternative to developing your own transformer.

You can follow Andrej Karpathy tutorial for a NanoGPT here: YouTube and use his GitHub with the code here: NanoGPT or you can develop your own code if you want.

The only requirement is to standardise your training data. Make it the same across your implementations.

3 Task 1.3

Comparison

Here, it would be best if you did a comparison of both models; you are requested to modify your Chatbot to use the same data as the transformer and answer the following:

- Compare the performance of the two models and explain in which scenarios you would prefer one over the other.
- How did the two models' complexity, accuracy, and efficiency differ? Did one model outperform the other in specific scenarios or tasks? If so, why?
- What insights did you obtain concerning data amount to train? Embeddiutilizedised? Architectural choices made?