Transformers

MCDA5511 Assignment #4

Due Nov 23 before midnight

Set Up

In this assignment you will implement a basic transformer from scratch to perform a simple task. The purpose is to build in-depth understanding of the inner workings of the transformer architecture. As such, if you have previous experience with PyTorch, you should try to do this assignment from scratch. However, if you are still learning, you can use GPT-4 Code Interpreter to help you.

You will submit your answers in a Jupyter notebook (*.ipynb* file). Use markdown cells to create section titles, python cells for code, and markdown cells to answer written questions. Make sure that the layout of your code and written answers is easy to follow.

Homework

(1) Implement a transformer from scratch in PyTorch, i.e. without relying on high-level transformer-specific libraries such as HuggingFace's `Transformers`. Instead, use only the basic components available in the PyTorch framework, such as `torch.nn`. To keep things tractable, use fixed hyperparameters.
   - Begin by populating a markdown cell with an overview of model components you will implement and a brief description of what they do, e.g. token and position embeddings, attention, feed forward network, etc.
   - Next, implement each component as a class. Where possible, include a test to check that outputs returned for each method have the correct shape.

(2) Train your transformer to reverse a sequence of five or less integers from 1 - 9. For example, if you input [3, 5, 2, 4, 1], the output should be [1, 4, 2, 5, 3]. Or if you input [9, 2] it should return [2, 9]. Your code should include the following steps:
   - Generate a synthetic data set of inputs and outputs.
   - Tokenize the data.
   - Create a training loop.
   - Write a function to generate output from the trained model.

   NOTE: If you wish, you may train the model to perform a more complex task.

(3) Draw a diagram showing the architecture you have implemented, including the dimensions of inputs/outputs at each step.

(4) Evaluate your model's performance quantitatively and write some commentary explaining your results. If you decided to build a model to do something more complex than reversing a sequence, you will need to give some thought to how you measure performance.

(5) Optional bonus question: Your model from previous steps should generate an output from the vocabulary (integers from 1 – 9), but it might not generate the correct answer (a reversed sequence). Spend some time manipulating hyperparameters and model architecture to see if you can get it to work. Add a markdown cell explaining what changes you made to improve the model results, and revisit the previous question with the improved version of your model.