Mid Term Project

DS-7339 Intro Gen AI for Industry Apps

1. **Goal:** Your goal with this mid-term project is to pre-train a smaller version of Qwen3 from scratch using a portion of the Tiny Stories dataset. Afterwards you will train a sentiment classifier on the last hidden layer’s embedding model (of dimension d\_model) on a sentiment data set provided. (*Note: In this project you will have to make design choices since you will likely be limited on CPU/GPU capabilities. Thus, it will important that you choose sizes of models, and data sets to illustrate the concepts below, but not too large that it takes too long to compute. Also, since we have combined the training model and classification in this project we will not have a Project 1*).
   1. **Reference code and dataset:** 
      1. Qwen3 reference model, use <https://github.com/rasbt/LLMs-from-scratch/blob/main/ch05/11_qwen3/standalone-qwen3.ipynb>
      2. Get the Tiny Stories data set from hugging face
      3. The reference dataset for sentiment scoring will be: Emotions\_classified.csv (we will upload this in the SMU class).
      4. (FYI only, you may look at the Gemma3 code provided in class that did a similar exercise by training it from scratch with Tiny Stories dataset).
   2. **Deliverables:** 
      1. **Qwen3\_small:** Modify the reference model and create a “Qwen3\_small” version to have approximately between ~100M-200M parameters (this will make it easier to train and use for classification). (show this in the code submitted **10 points**)
      2. **Tokenize & Train:** DO NOT use the “load pre-trained weights, or Load Tokenizer” but
         1. **Create or use a tokenizer (**e.g. you can use a simple word based tokenizer) (**10 points**) and
         2. **Train your Qwen3\_small**, on a subset of Tiny Stories (e.g. try 10,000-50,000 stories (100K stories is suitable for A100 GPU), with a train/test split of either 90%/10%, or 80%/20%).
            1. Show the loss function decreasing step-by-step and through at least two validation steps (**20 points**)
      3. **Test the model:** Use some prompts to show generation from the model just created. For example: (**20 points**)

📝 Testing prompt: 'Once upon a time'

----------------------------------------

Stopped after maximum 30 tokens

Temperature 0.5: 'once upon a time there was a little girl called lucy. she was three years old and she was very happy. she liked to play with her friends. one day, she went to the'

Stopped after maximum 30 tokens

Temperature 0.8: 'once upon a time there was a little girl called lucy. she was three years old and she liked to play in the forest to go on the park and cook and asked, "why'

Stopped after maximum 30 tokens

Temperature 1.0: 'once upon a time there was a cool girl. she was 3 years old. every saw a big brother on the field and climbed drawing on it. he flew up and saw lots of'

Stopped after maximum 30 tokens

Temperature 1.2: 'once upon a time there were languages baby thirsty! and a big buildings was cool away, but harmless.â€ infant. it would see, sam her carefully. one day, mia was big. "can tidy too found'

* + 1. **Embedding layer hook:** Create a “hook” before the final layer you can use to create a classifier (**15 points**)
    2. **Train a classifier:** on at least 1000 examples from the emotions\_classified.csv data set.
       1. Show the accuracy (**15 points**) {should be > 50%}
       2. Show the precision, recall, f1 (**5 points**)
       3. Show a confusion matrix (**5 points**)

Example (note: your sizes may vary):

📊 Loading emotions\_classified.csv dataset...

✅ Loaded 16000 samples from emotions dataset

📈 Dataset Overview:

Total samples: 16000

Positive samples: 7238

Negative samples: 8762

🎯 Selected samples:

Positive: 500

Negative: 500

Total: 1000

📝 Sample data:

1. 'i said i wanted to give you a little sample of the writing i...' -> joy -> positive

2. 'i feel a lot of positive intention behind it...' -> joy -> positive

3. 'i feel that there is a clever caption in the making here but...' -> joy -> positive

4. 'i pushed the feeling aside and contented myself with an appl...' -> joy -> positive

5. 'i feel that i have lived long enough i am leaving you with y...' -> joy -> positive

🔗 Extracting embeddings using model hook...

Extracting embeddings...

Processed 400/1000 samples...

Processed 800/1000 samples...

✅ Embeddings extracted!

Embeddings shape: (1000, 768)

📊 Train/Test Split:

Training samples: 800

Test samples: 200

Training positive: 400

Training negative: 400

🤖 Training sentiment classifier on emotions data...

📊 RESULTS:

Accuracy: 0.7050 (70.50%)

🎯 Confusion Matrix:

True Negatives: 78

False Positives: 22

False Negatives: 37

True Positives: 63

📈 Detailed Metrics:

Precision: 0.7097

Recall: 0.7050

F1-Score: 0.7033

📊 Per-Class Metrics:

Negative (0):

Precision: 0.6783

Recall: 0.7800

F1-Score: 0.7256

Support: 100

Positive (1):

Precision: 0.7412

Recall: 0.6300

F1-Score: 0.6811

Support: 100

Precision: 0.7412

Recall: 0.6300

F1-Score: 0.6811

Support: 100