
EECS 182 Deep Neural Networks

Fall 2022 Anant Sahai Final Review: Transformers, Finetuning, and Prompting

1. Transformers

- (a) Let's say you are training a navigation robot using a robotic navigation task. The user describes in language an object in the house, and the robot must navigate the house to find it. The robot sees an image from its front-facing camera at each timestep, and it outputs one of three movement actions: [LEFT, RIGHT, FORWARD]. You have a dataset of example robot trajectories. Describe how you would use a transformer for this task. (What architecture? What is inputted to the transformer? What is outputted?)

Robotics application, how use transformer for this?

- (b) Why are subword tokenizers (e.g. WordPiece encodings or byte pair encodings) preferred over word tokenizers?

2. Finetuning

- (a) If you pretrain using a masked autoencoder, when you finetune the autoencoder encoder for downstream tasks, do you still mask the inputs?
- (b) Let's say you want to train an LSTM encoder/decoder model on translating from English to Spanish using a paired English/Spanish training set. You also have a much larger corpus of unpaired English sentences. Describe one way to pretrain the LSTM encoder using the unpaired data.

- (c) For each of the following finetuning problems, describe whether you should prefer to use (a) feature extraction (also called linear probing), (b) full finetuning, (c) hard prompting, or (d) soft prompting.

- (a) You are using a 175B parameter language model for a question-answering task. You have a dataset of 100k examples.
- (b) You are using a 90B parameter language model for a spam classification task. You have a task description but no training data.
- (c) You are using a 1B parameter conv net pretrained on ImageNet for wildlife classification task. You have 100 training examples.
- (d) You are using a 1B parameter vision transformer pretrained on ImageNet for an X-Ray fracture localization task. You have 100M training examples.

3. Prompting

- (a) Typically, when you create a soft prompt for use with a GPT model, you prepend the prompt to the left of the input. Could you also get good performance by appending it to the right?
- (b) Let's say you would like to use hard prompting with a large GPT model. You have a dataset of 10 thousand training examples for your downstream task. Would it be a good idea to include all of these examples (except for a held-out validation set) in your prompt?