Kenneth Wang
Qingyuan Liu
Kevin Cai
Sewon Sohn

EECS 182 Project: Multimodal Chain-of-Thought Reasoning Model Commentary for Initial Submission

This project built a homework assignment (option 1) that is based on the Multimodal Chain-of-Thought reasoning paper (Zhang et al. 2023, linked here). Essentially, this homework assignment breaks down the paper and simplifies the model into sequential steps, taking students from the beginning with the ScienceQA dataset (which consists of multiple-choice elementary natural science questions) and pre-processing the raw data to the end (running the model and seeing the improvements in the rationales). As such, the high level goal of this project and this homework assignment was to expose students to the two main key ideas while also giving them hands-on, practical experience in implementing part of the model themselves. In addition, there are multiple connections to EECS 182 course content and smaller ideas briefly touched upon and explored throughout the notebook, written questions, and solutions.

Chain of thought (CoT) refers to the mental process of reasoning and inference that humans use to arrive at an answer or solution to a problem. It involves synthesizing information from multiple sources, making logical connections between ideas, and integrating them into a coherent line of reasoning. In the context of natural language processing and deep learning models, CoT prompting involves generating intermediate reasoning steps to arrive at the final answer to a question. This process allows the model to break down complex questions into simpler, more manageable steps, and to leverage information from multiple sources, such as text and images, to arrive at a more accurate answer.

The Science Question Answering (ScienceQA) dataset comprises a total of 21,000+ multiple-choice science questions sourced from elementary and high school curricula. Through the attached notebook, students will explore a subset of this dataset, which consists of questions that only contain a text context as well as questions that have both text and image contexts.

The main two ideas in this paper are the multimodal nature as well as the chain of thought reasoning imposed by the two-stage reasoning framework. The first two parts, Prompt Building (part a) and Add Images (part b) expose the students to the intuition behind the model, without actually exposing what the model architecture is. The first subpart of Prompt Building has a question (and related code in the notebook) that exposes the ScienceQA dataset and shows how the "solution" attached to each question can help the student (or any other reader) easily reach the desired answer. As such, this suggests the solution could be a target that the rationale generation model can be trained towards. The Add Images part does a similar thing, where it first lets the student look at a sample question without the image context, and then has them look at the same question with the additional image context. Through this part, the student should naturally come to the conclusion that the image context helps them (and any other readers or reasoners) figure out the answer. In this manner, this project hopes that students will come about to the two main concepts of multimodality and chain of thought naturally without needing to be "told" that this is a better model.

Prompt building is the first part of the code that the students will conduct in this homework assignment. This process is commonly used in tasks such as text completion, translation, and question answering, where the model is required to generate output that is consistent with a given input prompt. Prompt building can be a highly effective technique for improving the accuracy and performance of language models, as it allows us to fine-tune the model's behavior for specific tasks and domains. Additionally, prompt building can help to mitigate the problem of bias in language models, as it provides a way to explicitly specify the desired output and constrain the model's behavior. Students will be asked conceptual questions on prompt building in order for them to comprehend the data through each step of a complex, transformer-based language model that has multiple parts.

The next part of is dataloading. Dataloading typically involves reading data from one or more sources, such as a file or a database, and performing preprocessing steps such as normalization, transformation, or augmentation, in order to prepare the data for use in the model. This process determines how efficiently and effectively the model can learn from the data. We show students how the tokenizer works by asking the students to fill out part of the code in the data loader, which is used by the pytorch style trainer to retrieve batches of data for training, evaluation and testing. Furthermore, the students can visually see the data before being tokenized

(the raw text that makes sense as questions and solution contexts) as well as the data after tokenization (simply numbers).

The students will then implement the encoder-decoder architecture model. The written part explains the model architecture and the coding part makes the students code up a few of the layers, allowing them to deeply understand the underlying architecture behind the model. The students have to determine the exact ways to process prompts and add paddings and output tokens, labels and attention masks. Through this, the students practice their comprehension of model architecture and develop further coding skills, albeit with heavy hints and much of the work done for them already. Through this manner, we hope the students will engage further with the encoder-decoder architecture model and also understand how the selective attention mechanism is implemented by doing the coding part.

At the end, the students get to see the rationales generated, further reinforcing the chain of thought model they were introduced at a more intuitive level at the beginning.

This homework assignment also applies the idea of the model to other concepts learned in class. One of them is the model's resemblance to ResNet skip connections. The two-stage framework takes the input (text and image), generates rationale, and then appends the rationale to the original input to create a modified input. This modified input is then passed into the second model, the inference model. Another concept is zero-padding. When we generated the rationale before it was trained, the rationale consisted of random repeated words that filled the last few lines. We deduced that this is because the text input is zero-padded. Finally, although this is extra and only touched in the solution, we enable a potential student to explore the idea of multimodality further and see how it might be considered as regularization. In this solution, we tie back multimodality to the course concept of regularization, with a brief comparison to feature augmentation of OLS being equivalent to ridge regression. Of course, this isn't fully explored but simply left as a tidbit that students reading through the solutions can explore more on their own.