Group 11 Approach

Data preprocessing and problem approach

In this project, we framed the task as an open-ended question-answering problem using the TGIF-QA dataset. The dataset was initially downloaded and underwent a preprocessing phase, where we organized the diverse question types into a structured format, creating a dataframe suited for open-ended question answering. The questions in the **TGIF-QA** dataset span various categories, and preprocessing them ensured that the data was aligned for input to our model. A subset of the GIFs from the dataset was downloaded using asynchronous methods to expedite the process and efficiently handle the large number of media files involved.

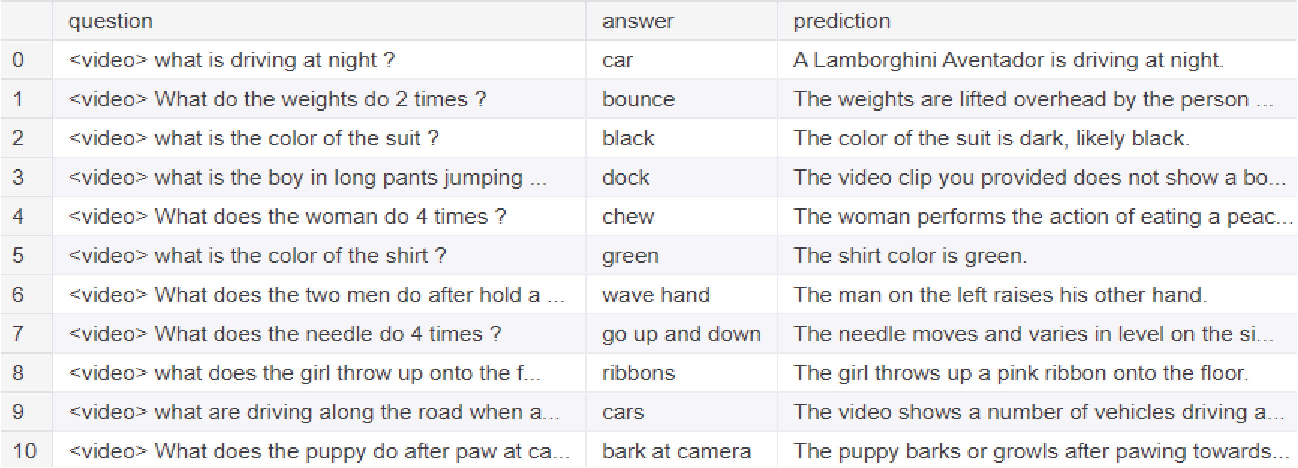
Our subsequent literature review explored multiple approaches to video question-answering, leading us to select the **Tarsier** model for implementation. This decision was based on its strong track record in similar tasks and its architecture, which leverages both video and text data. We devoted a significant amount of time to comprehensively understanding the Tarsier implementation before initiating any experimentation.

Modeling: Tarsier for Zero-Shot Inference

The Tarsier model incorporates advanced techniques for multi-modal learning. It employs a **CLIP-ViT encoder** to generate embeddings from video frames, which are then combined with text embeddings derived from the question. These inputs are passed to a large language model (**LLM**) trained to **understand temporal relationships** within the video, making it particularly well-suited for complex question-answering tasks that require sequential frame analysis.

To facilitate the integration of the TGIF-QA data with the Tarsier model, we first developed a custom Dataset class. This class efficiently processed the data and prepared inputs in the format required by the model. Once the model was configured, we employed the bits-and-bytes library for proper **quantization** of our model. This method allows the model to use fewer computational resources with marginal loss in precision, making it more efficient for inference on large-scale datasets.

After setting up the model, we proceeded with zero-shot inference. Initial human evaluations of the outputs suggest that the model is performing well, providing promising results in terms of both accuracy and relevance to the questions posed (please refer to the figure below).



	question	answer	prediction
0	<video> what is driving at night ?	car	A Lamborghini Aventador is driving at night.
1	<video> What do the weights do 2 times ?	bounce	The weights are lifted overhead by the person ...
2	<video> what is the color of the suit ?	black	The color of the suit is dark, likely black.
3	<video> what is the boy in long pants jumping ...	dock	The video clip you provided does not show a bo...
4	<video> What does the woman do 4 times ?	chew	The woman performs the action of eating a peac...
5	<video> what is the color of the shirt ?	green	The shirt color is green.
6	<video> What does the two men do after hold a ...	wave hand	The man on the left raises his other hand.
7	<video> What does the needle do 4 times ?	go up and down	The needle moves and varies in level on the si...
8	<video> what does the girl throw up onto the f...	ribbons	The girl throws up a pink ribbon onto the floor.
9	<video> what are driving along the road when a...	cars	The video shows a number of vehicles driving a...
10	<video> What does the puppy do after paw at ca...	bark at camera	The puppy barks or growls after pawing towards...

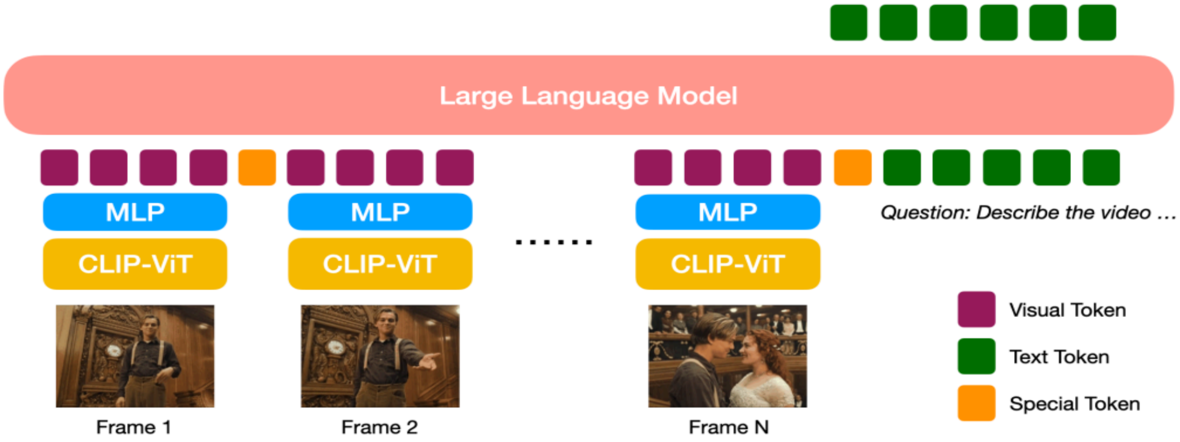
Please refer to this notebook for the code.

Fine Tuning

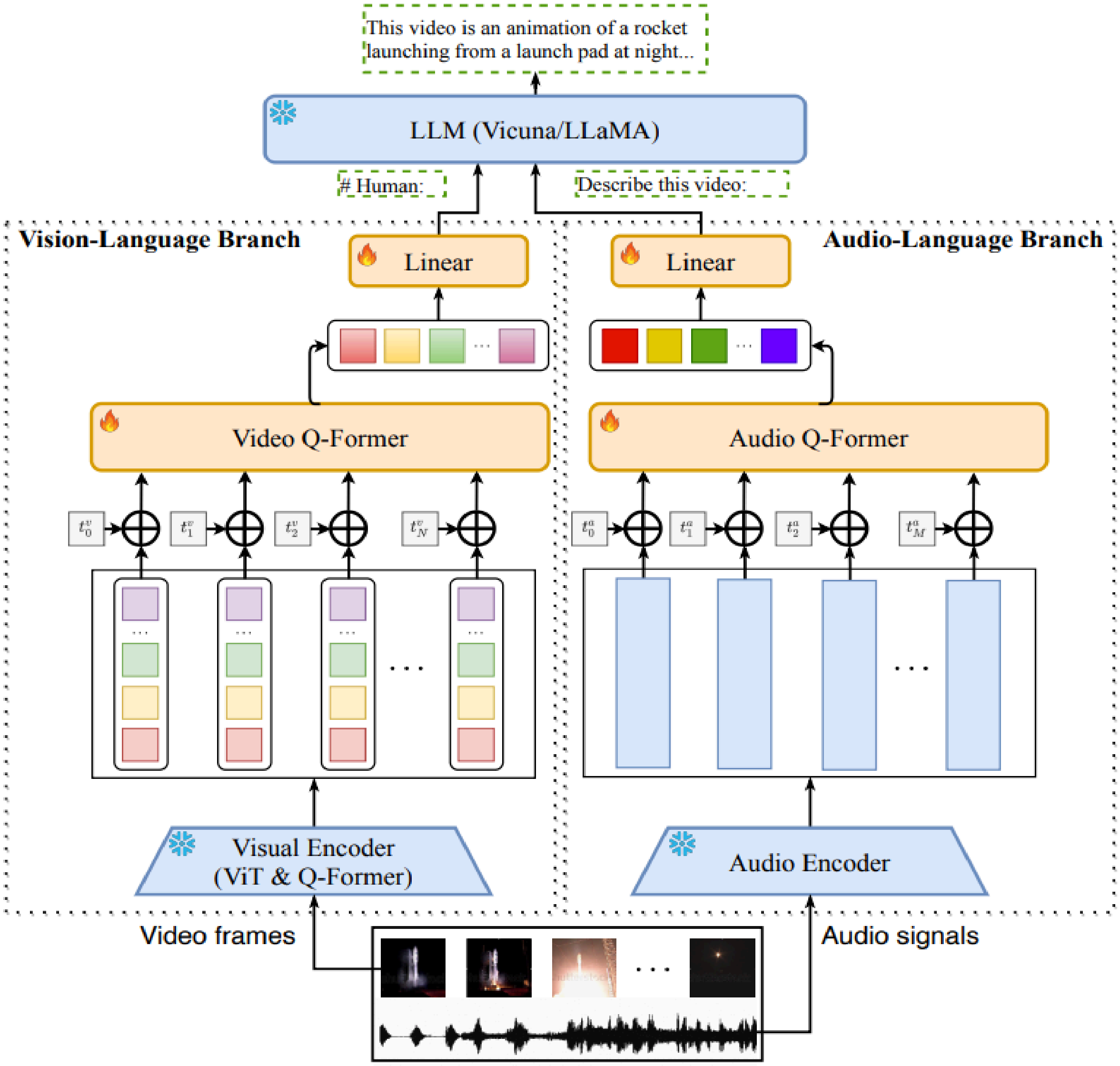
Initially, we attempted to fine-tune the model using **Q-LoRA**, a low-rank adaptation method designed to efficiently fine-tune large models by introducing trainable low-rank matrices. However, we encountered issues when implementing this with the **Tarsier** model, which led to errors during the process. As a result, we shifted our focus to researching how other visual question answering (VQA) models handle similar tasks, aiming to gain insights and find alternative approaches for improving our model's performance.

Tarsier

According to the paper, **Tarsier** processes **GIFs** and **videos** by dividing them into individual **frames**, selecting a **fixed number of frames** for analysis. Each frame is then passed through **CLIP** to obtain an **image embedding**. Since the image embeddings are not directly aligned with the **LLM's** embedding space, a **multi-layer perceptron (MLP)** layer is applied to transform the image embeddings. The output of the MLP layer, along with the corresponding **text embeddings**, is then fed into the **LLM** for further processing.



Video LLAMA



In the **Video LLaMA** paper, the authors incorporate **Q-Former** (introduced in **BLIP-2**) to effectively manage **temporal information**. Other aspects of the model remain similar to **Tarsier**.

Our approach

Image Encoding

For **image encoding**, we utilized a **pretrained BLIP model** to extract image features. We specifically obtained the output from the **Q-Former** component of the BLIP model to derive these features.

Temporal Features

To capture **temporal features**, we employed the **Q-Former** that has been trained on the provided dataset. This allows us to effectively handle the temporal dynamics present in video data.

Projection Layer

To ensure that the image embeddings are compatible with the Q-Former, we introduced a **projection layer**. We applied a similar approach to the temporal features to align their dimensions appropriately.

Language Model (LLM)

For the text processing component, we selected **LLaMA 3.2**. We simply prompted the LLM with the temporal features along with the questions to generate responses.

Why LLaMA 3.2?

LLaMA 3.2 is a lightweight version that requires less GPU memory, making it easier to load and run on platforms like **Kaggle**.

Why BLIP-2 for Image Encoding?

Given the limited time available to train a full model from scratch, we opted to extract image embeddings from BLIP-2. We chose BLIP-2 because it is lightweight compared to other visual encoders. Additionally, we wanted to leverage the Q-Former for capturing spatial information, which BLIP-2 already integrates, thus effectively addressing our requirements.

Scope of improvement

Due to time constraints, we only utilized the output from the second Q-Former. However, incorporating the output from the first Q-Former could enhance our results by providing additional spatial features. Including both outputs would allow for a more comprehensive representation of the image data, potentially improving the model's performance.

Research and references

- Video-LLaMA An Instruction-tuned Audio-Visual Language Model for Video Understanding
- LLama Paper introducing llama models
- Tarsier Recipes for Training and Evaluating Large Video Description Models
- BLIP-2 Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models
- Video-ChatGPT Towards Detailed Video Understanding via Large Vision and Language Models