BU AI Tour Assistant

Weining Mai, Andy Yang

February 13, 2024

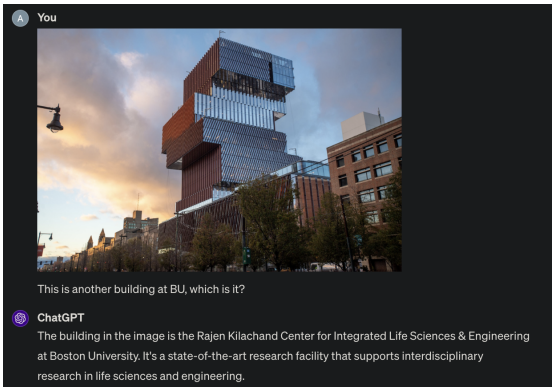
Abstract

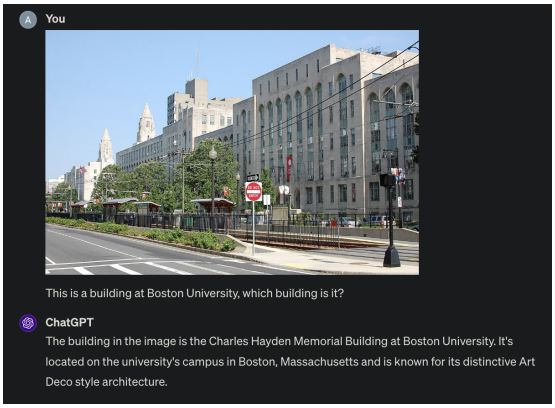
In today's digital era, we are often struggle to find the desired location with only an image from internet. Leveraging the **open source** multi-modal LLaVA model, our project transforms street-view images into vision chatbot that answer questions regarding BU's campus, unlocking new potentials for location-based services and virtual tour guide.

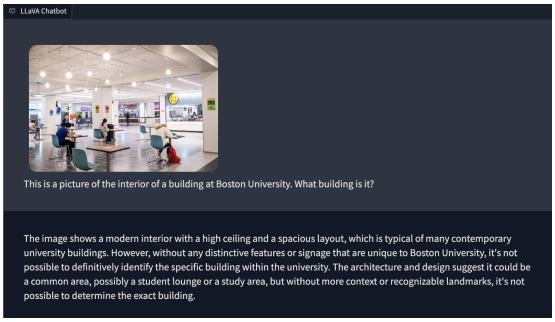
GithubRepo

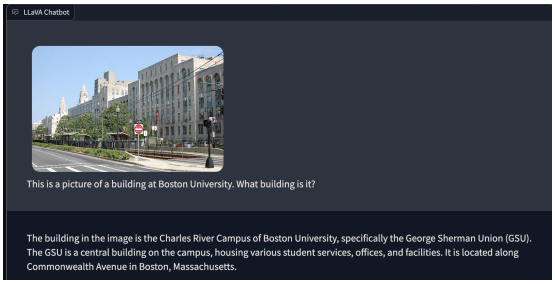
Introduction

Everyday, countless of photos are taken, many of which contain valuable spatial information that, if properly harnessed, can revolutionize how we interact with our environment. From improving emergency response times by quickly identifying incident locations to enhancing travel experiences with context-aware recommendations and photoshoot spots, the stakes are high. However, despite advancements in technology, the challenge of accurately classifying the geographic location of an image remains significant due to the complex variability in landscapes, urban settings, and the sheer diversity of geographical features worldwide. Addressing this problem not only bridges the gap between digital and physical realms but also innovative location-based services that can adapt to our needs in real-time. Our project aims to tackle this challenge head-on, using the lLava model to turn images of BU's campus into detailed descriptions of the location, thereby making the world not just more connected, but also more navigable and understandable for new students and tourists. Currently, both lLava 34b and GPT4 are not very capable of classifying BU buildings like CDS, CAS, and GSU:









Related Work

What really inspired this project is the open-source version of GPT4-vision, that is, lLava. [2]. It has the potential of giving back the location of the image with building names and states. We learned that the Boston University campus can be difficult to navigate for prospective students or tourists. Frequently, I was asked where is this building located. Although GPT4 or lLava are able to predict location on a larger scale, they might not be able to guess the exact location of buildings. We learned that various parameters of lLava gave different answers. Bigger parameters gave a better and more precise description of the location, whereas a 7b parameter gave a generic one. Furthermore, both lLava and PIGEON [1] make use of CLIP developed by OPENAI. We see that PIGEON [1] is already capable of predicting locations within some radius and beating top-rank guessers. However, due to privacy concerns and the ethical use of AI, PIGEON has not released the code behind their solutions.

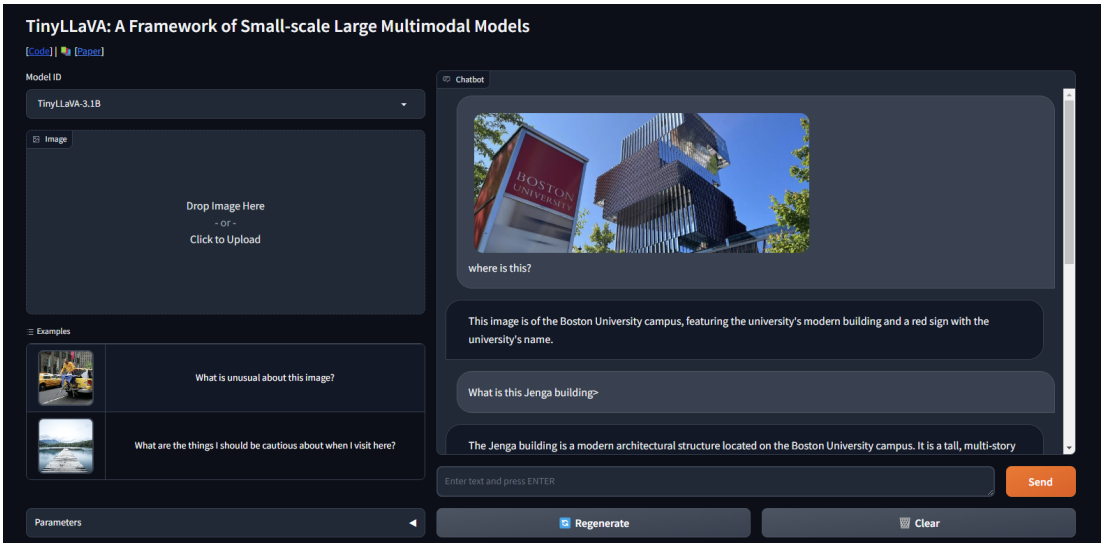
Recent developments with LLaVA have led to TinyLLaVA, a smaller version (Phi-2) offering comparable performance with far fewer parameters[3]. Given our limited computing resources, we plan to fine-tune TinyLLaVA on text-answer BU image data. TinyLLaVA offers several advantages over the original LLaVA: it's easier to modify and refine with

smaller datasets, requires less powerful GPUs for training, processes and generates text faster, and is more accessible due to its lower costs. These factors enable us to innovate more quickly.

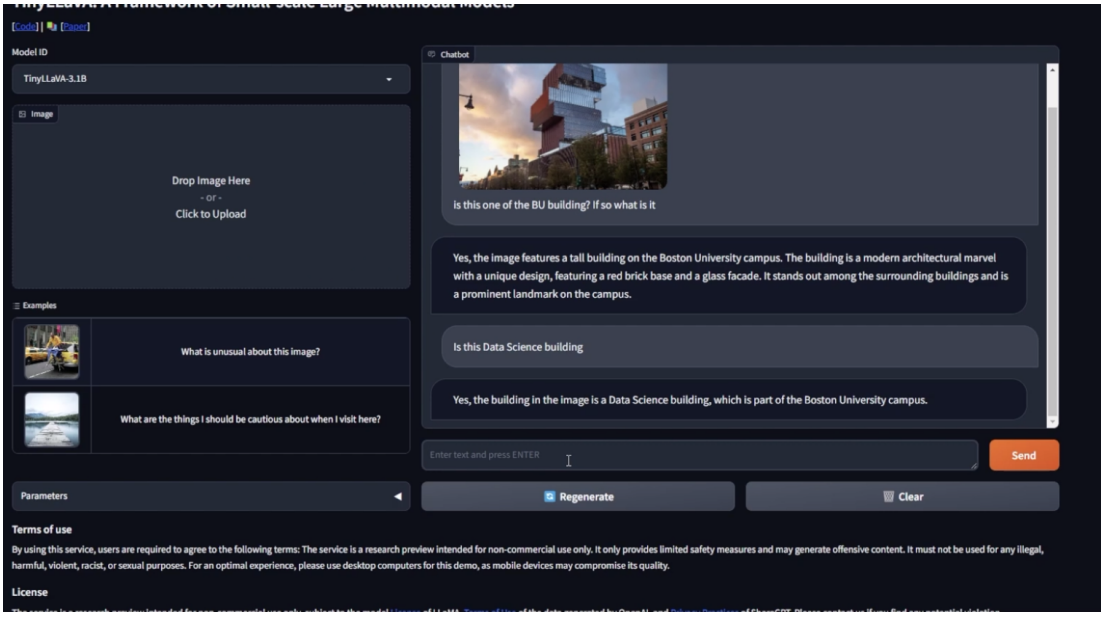
Proposed Work

Our proposed solution leverages the open-source TinyLLaVA model to provide specific answers about Boston University (BU) based on images. Since the original LLaVA lacks specific knowledge of BU, we'll fine-tune TinyLLaVA on images and corresponding textual descriptions to enhance its understanding. While ambitious, we'll start with the achievable goal of accurately identifying BU buildings in images. We also aim to fine-tune TinyLLaVA for consistent responses, minimizing overly creative outputs. The smaller size of TinyLLaVA helps reduce the likelihood of hallucinations. We've chosen the TinyLLaVA-3b model for its accessibility and reduced parameter count. To facilitate the fine-tuning process, we'll utilize Runpod's GPU servers and a Gradio interface on Google Colab. Due to our limited dataset, we'll employ the **PEFT** method, which updates only a subset of TinyLLaVA's parameters. This allows us to generate smaller checkpoints and work with less powerful GPUs, enabling faster iteration.

Below is an example of inference with the finetuned model on Gradio. We see that the model is capable of giving domain-specific answers such as knowing what "Jenga building" is. By going through the motion, we now would like to obtain more data and fine-tune again. Both GPUs A6000 and A100 have been experimented with and have been successful.



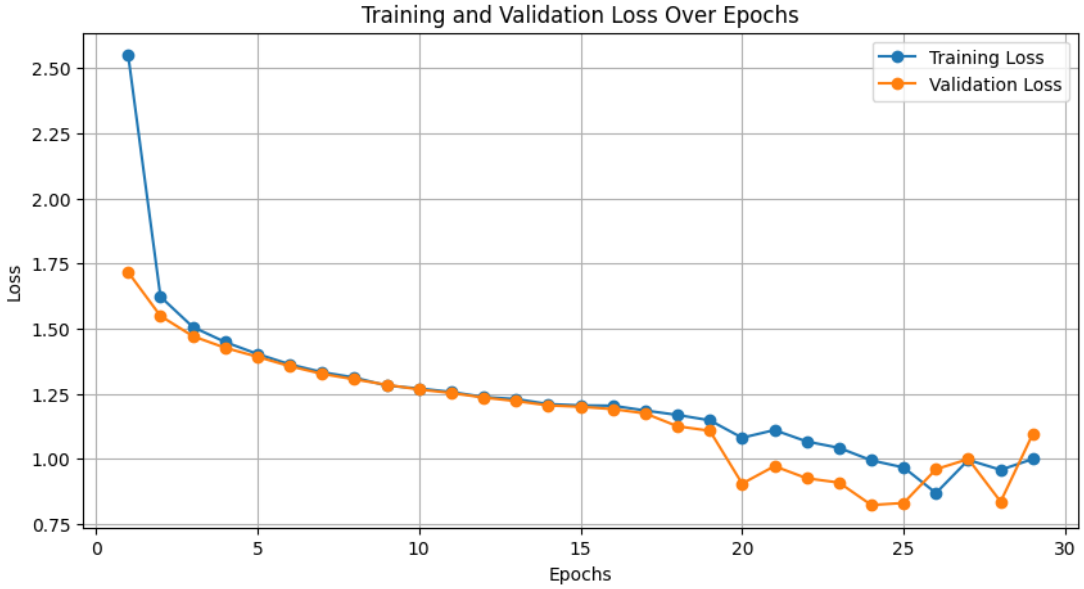
Below is an example of our inference on the most recent up to date model



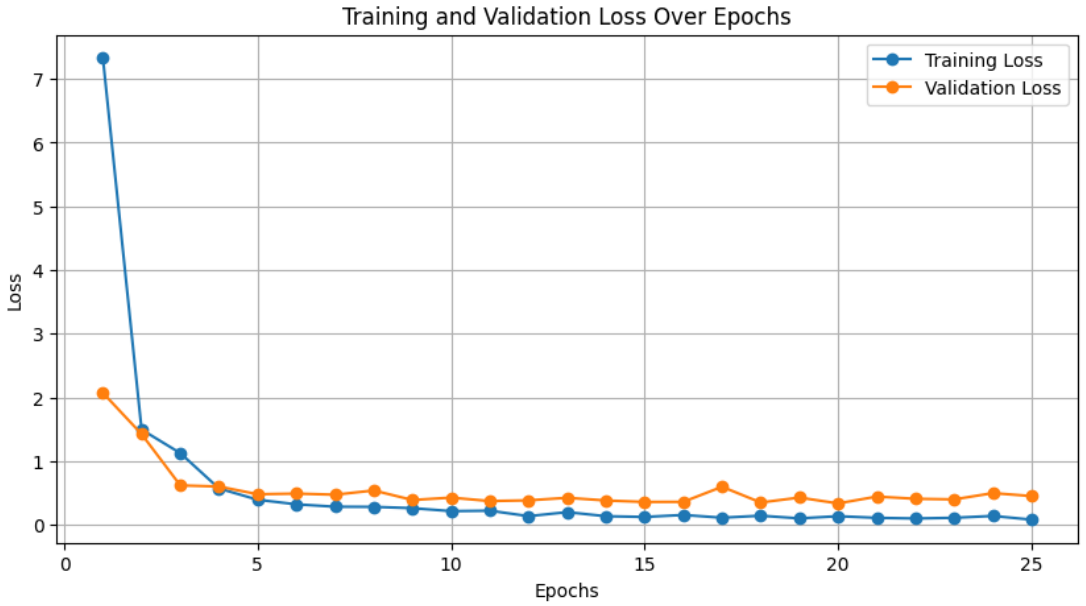
Additionally, we created several image classification models trained on our dataset. Although not initially a part of our project proposal, we were motivated by curiosity to see if we could build a successful classifier after we collected our images.

We started with a very small baseline convolutional neural network, created using Tensorflow. This model consisted of two convolutional layers and two max pooling layers, in alternating order for feature extraction. After flattening the output of the convolutional blocks, the network includes a dropout layer set to 0.5 to try to reduce overfitting, and finally, the dense layer with 3 outputs and softmax activation. This model achieved a validation accuracy of 0.6522 after training for 15 epochs. See the graph of training and validation loss below.

For our next model, we added made our network deeper, and added some other features. Specifically, we now had four convolutional/maxpooling blocks, three different dropout layers, and l2 regularization in two dense layers. This new architecture performed a bit worse, achieving a validation accuracy of 0.6087 after 29 epochs. Both this model and the baseline were compiled using Adam optimizer, and categorical cross-entropy loss.

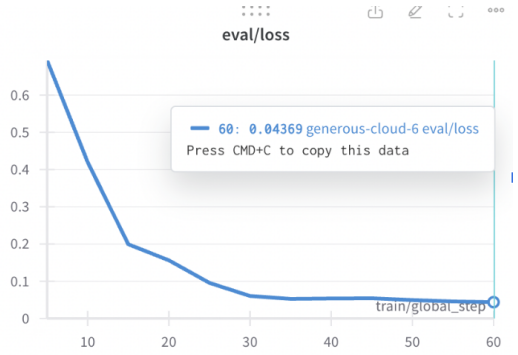


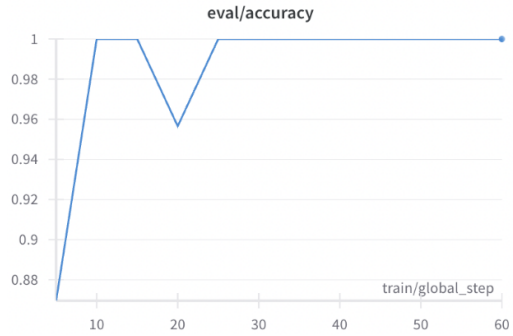
We also tried transfer learning with VGG16, which is a larger network and also trained on imagenet. After freezing the layers in this VGG16 model, we added our own two dense layers and dropout that would be trained on our dataset. This approach was pretty successful, giving us a 0.8696 validation accuracy at the end of 14 epochs (early stopping).



Here are the test set predictions:

Lastly, we also finetuned a vision transformer on our dataset, which performed the best. We chose the `google/vit-base-patch16-224-in21k` model, which is trained on ImageNet-21k. This achieved the highest accuracy, hitting 100% after around 10 steps.





Feel free to take a look at the training/fine tuning notebooks in the repo in the `image_classification` directory. Overall, although this wasn't the main focus of our project we found it pretty interesting to apply some topics that we learned about in class to our project, and got some good results while doing so.

Datasets

We plan to collect our own dataset of exterior and interior pictures of CDS, CAS, and GSU. To start, there are lots of pretty good images that can be found online. Google Images is a good source for exterior shots of buildings and BU has images of most classrooms that can be web-scraped, for example: https://www.bu.edu/classrooms/classroom/cds-164/.

We also plan to collect some images of our own to ensure a comprehensive, diverse, and plentiful dataset. Specifically, we anticipate that we will need to take pictures of hallways, stairways, foyers, and location-unique features like the Spark space and CAS think tank.

After image collection, we will need to label our data. For each image, we need question/answer pairs. For example, potential questions could be about the building's name, current uses, architectural details, and historical significance. Answers to these questions would need to be informative and accurate, as they are crucial for training the model to understand and generate the correct responses.

We expect we will need at least 150 images to train a baseline model with ideally 10 per building, and around 10 per feature (exterior, classroom, hallways, etc). This will be our first deliverable/checkpoint, which we hope to complete in 2-3 weeks.

In our original plan, we aimed to have a complete dataset of images by this time. However, we've adjusted and refined our scope based on our progress and insights thus far. Rather than dedicating our initial efforts to creating a comprehensive collection of images and captions, we pivoted towards the development of an initial, simple model. We think this is a better approach since it allows us to iterate quicker and test the viability earlier on. So far, we've seen success with this approach and we still expect that by the semester's end, to have successfully compiled our own comprehensive dataset along with a (better) fine-tuned model.

Final Update: We created a dataset of 120 image/caption pairs across three classes: CDS, GSU, CAS. This proved sufficient for our needs, but we do think that better results can come from more diverse and consistent captioning of images. An example of an image/caption pair is provided:

"The image shows a close-up view of the Data Science Building at Boston University. This striking piece of architecture is characterized by its unique copper-colored metal fins that create a textured facade. Below the layered fins, the lower levels are encased in large glass windows that reflect the surrounding buildings and skies, providing a sense of openness and blending the inside with the outside. The building has an overhanging section which seems

to float above the ground level, creating a covered area that might serve as an entrance or public space. A crowd of people, possibly students and faculty, is seen crossing the street in front of the building, indicating the dynamic and bustling atmosphere typical of a university campus setting. The overcast sky suggests a gloomy or chilly day in the city of Boston."

Evaluation

To evaluate the performance of fine-tuned TinyLLava outputs, we can curate a separate test set of images and evaluate the outputs by comparing the base model with the fine-tuned model. We hope to see if fine-tuned TinyLLava BU QA can give back more accurate information about BU than the not-funetuned one.

Here we have few metrics for evaluating data or the model
1. Answer the question using a single word or phrase(ideally model output correct BU building's name)
2. performance metrics: if we want to deploy our model, we would like to consider the inference time, storage space, and GPU usage.
3. Inspecting quality of outputs through human judgment.
Another thing to keep in mind is if the model can answer related questions consistently. For example, if it identifies a building correctly, it should also accurately answer more detailed questions about that building and not hallucinate or become sidetracked. Logical reasoning can also be tracked. For questions that require piecing together information (e.g., "What type of classes are held in this building?"), we should monitor the model's reasoning and check if its answers make sense given what it knows.

Timeline

1. **Weeks 1-3:** Data collection
2. **Weeks 4-5:** First model
3. **Weeks 6-9:** Make improvements
4. **Weeks 10-11:** Expand to more buildings

Conclusion

Our project, aims to leverage the tinylLava model for identifying and providing detailed information about Boston University campus buildings from images. By creating a specialized dataset of campus images and corresponding question-answer pairs, we look to enable

the model to not only recognize buildings but also to provide users with relevant details about them.

This endeavor highlights the potential of visual question-answering systems to make physical spaces more navigable and understandable through digital means. Our work demonstrates that with careful dataset preparation and model fine-tuning, it is possible to enhance location-based information retrieval in ways that could benefit things like tourism and education.

To further enhance the capability, we need a better quality of data such as more conversations or annotate images with object detection for an accurate visual QA about BU campus. Because it is open source and small model, others have the chance to finetune further to fit in their use cases with limited compute resources.

References

[1] Lukas Haas, Michal Skreta, Silas Alberti, Chelsea Finn. *PIGEON: Predicting Image Geolocations* . arXiv, 2023.

[2] Haotian Liu, Chunyuan Li, Yuheng Li, Yong Jae Lee. *Improved Baselines with Visual Instruction Tuning.* arXiv, 2023.

[3] Baichuan Zhou, Ying Hu, Xi Weng, Junlong Jia, Jie Luo, Xien Liu, Ji Wu, Lei Huang. *TinyLLaVA: A Framework of Small-scale Large Multimodal Models.* arXiv, 2024.