BIODS 276 / CS 286 Advanced Topics in Computer Vision and Biomedicine
Fall 2024

Course Project

The goal of the project is to gain hands-on experience interacting with some of the large vision and vision-language models discussed in class. You may choose among multiple options for the project type, described in further detail below. The intent of all project types is to gain a deeper understanding of the capabilities, limitations, and opportunities for future advancement of these models, whether through comparative analysis, investigation of models for a particular biomedical application, or exploration of technical innovations to address existing limitations.

You may work individually or in teams of 2 on the project, and grades will be calibrated by group size. Your project may be related to that of another class project as long as permission is granted by instructors of both classes; however, you must clearly indicate in the project proposal, milestone, and final reports the exact portion of the project that is being counted for this course. In this case, you must prepare separate reports for each course, and submit your final report for the other course as well.

Following is a description of the project options you may choose from, the project components that you will need to turn in, and the grading rubric.

Project Options

1. **Comparative Analysis.** Conduct a comparative or “red teaming” analysis of at least two large vision or vision-language models, considering a biomedical use case or motivation. Here, comparative analysis refers to probing the models to analyze where they work well or not, and identifying weaknesses, vulnerabilities, biases, or failure modes, in order to better understand limitations and suggest areas for further research. Your project should include comparative analysis of your selected models, which can be two or more completely different models, or multiple versions from one model family. You may use both biomedical and non-biomedical data, but the project should involve at least some biomedical data.
2. **Implement an agentic system.** Implement an agentic system that incorporates large vision or vision-language models to address a specific biomedical problem. Here, “agentic” refers to developing a system where model “agents” are used (potentially out-of-the-box, or in API-based fashion) in combination to achieve a specific goal. For this project, you must incorporate at least two model agents in your system, but only one must be a large vision or vision-language model (the other could be a language-only model, a specialized object detection model, or a search engine, for example).
3. **Explore a technical innovation.** Explore and experimentally assess the effectiveness of a novel technical approach or innovation to enhance the capabilities of existing vision or vision-language models, considering a biomedical use case or motivation. For example, this could involve making model or architecture improvements, or curating a new instruction-tuning dataset. While the technical innovation does not need to be

biomedicine-specific, at a minimum the potential utility of the innovation for specific biomedical applications must be discussed, and some biomedical data must be used to assess the effectiveness of the innovation.

If you are not sure about the suitability of a potential project idea, please discuss with the course staff.

Computing Resources

Different projects may involve different levels of compute. Please consider this when choosing your project. Below, we provide a list of some resources that you can leverage for your project.

- GPUs: For projects that require GPU usage (e.g., for training models), all students in the class will have access to $50 GCP coupons to use Google Compute Engine for developing and testing your implementations. When you first sign up on GCP, you will also have $300 free credits. You can follow Google Cloud Setup and Tutorial to learn how to use it.
- Gemini 1.5 Flash/Pro API: The Gemini API "free tier" is offered through the API service with lower rate limits for testing purposes, and you can also play with the models in Google AI Studio.
- OpenAI API: You can use the gpt-3.5-turbo API for free with a limit of $100/month. The text embedding models and automatic speech recognition also have a free tier.
- This repo introduces more free LLM/VLM APIs you can use.
- In addition to APIs for higher-volume usage, you can also interact with some models for free through their online interfaces (e.g., GPT-4o mini is included in the free tier of ChatGPT).

Project Grading Breakdown

(10%) Project Proposal
(25%) Project Milestone
(5%) Project Advising Session with TAs
(10%) Project Final Presentation
(50%) Project Report

Project Proposal

Your project proposal is due on Wednesday, Oct 23. It should be a 0.5-1 page document excluding references and figures, using the NeurIPS template. The proposal should include the following (with grade breakdown):

- Title, Selected Project Option, Authors(s)
- (25%) What is the biomedical problem or motivation for your proposed project? Why is it interesting?
- (25%) Give a brief overview of your planned approach. Describe any risks and mitigation strategies.
- (5%) What large vision and/or vision-language models will be involved in your project (there must be at least one). Why did you choose this model?

- (20%) What biomedical data will you use in your project? Please include relevant characteristics such as the source and size of the data, and a sample of the data. Explain any potential obstacles and mitigation strategies.
- (25%) How will you evaluate your results? Qualitatively, what kind of results do you expect (e.g. plots or figures)? Quantitatively, what kind of analysis will you use to evaluate and/or compare your results (e.g. what performance metrics or statistical tests)? What is your hypothesis regarding the results you expect to see?
- Mention any required computational resources or APIs needed for this project, and how you plan to acquire them (e.g., lab GPUs, course Google Cloud credit).

Submission: Please submit your proposal as a PDF on Gradescope. If you are working in a team, only one person on a team should submit. Please have this person add the rest of your team as collaborators as a “Group Submission”.

Project Milestone

Your milestone is due on Wednesday, Nov 20. It should be a 3-4 page document using the NeurIPS template. It should include preliminary sections towards the final report, in standard formats of NeurIPS research papers. The milestone should include the following (with grade breakdown):

- **Title, Selected Project Option, Authors(s)**
- **(20%) Introduction.** Introduce the objective of your project, and the landscape for why this is interesting and what has been done before in this space. Discuss the biomedical problem or motivation. Describe your overall plan for approaching the work, what contributions you expect to make, and why this is interesting in the context of the described landscape.
- **(25%) Related Work.** Describe in detail existing work related to your problem, how they are related to each other, and how your work relates to these. We expect this to be comprehensive and thorough, and with at least 10 citations discussed and cited accordingly.
- **(15%) Data.** Describe in detail the data that you are using, including the source(s) and size of the data, relevant statistics, and qualitative examples if appropriate.
- **(20%) Approach.** Thoroughly describe the methods that you intended to use in your approach, and if applicable, baseline methods you plan to compare against.
- **(20%) Preliminary Results.** Describe preliminary results that you have obtained. You should have progressed in your project sufficiently to produce some preliminary outputs from a large vision or vision-foundation model. These could correspond to a subset of your proposed data, or to a baseline model, depending on your chosen project option. Present your preliminary outputs and corresponding initial analysis in this section. You should also describe anticipated next steps and any obstacles that have come up.
- Each of these sections will be graded on a scale of 0-3 based on how well each component of each section is addressed.

Submission: Please submit your proposal as a PDF on Gradescope. If you are working in a team, only one person on a team should submit. Please have this person add the rest of your team as collaborators as a “Group Submission”.

Project Advising Session

Between the project milestone and the project final report / presentation, we will schedule an advising session for each project group with course staff, to discuss and receive additional feedback. Attendance will comprise part of your total project grade. More details will be provided closer to the date.

Project Report

Your project report is due on Wednesday, Dec. 11 at 11:59pm PT. It should be a 7-9 page document using the NeurIPS template. The report should include the following sections (with grade breakdown):

- **Title, Authors(s)**
- **Abstract.** A paragraph overview of the goal, approach, contribution, and key results.
- **(15%) Introduction.** Introduce the objective of your project, and the landscape for why this is interesting and what has been done before in this space. Discuss the biomedical problem or motivation. Describe your overall plan for approaching the work, what contributions you expect to make, and why this is interesting in the context of the described landscape.
- **(15%) Related Work.** Describe in detail existing work related to your problem, how they are related to each other, and how your work relates to these. We expect this to be comprehensive and thorough, and with at least 10 citations discussed and cited accordingly.
- **(15%) Data.** Describe in detail the data that you are using, including the source(s) and size of the data, relevant statistics, and qualitative examples if appropriate.
- **(20%) Approach.** Thoroughly describe the methods that you intended to use in your approach, and if applicable, baseline methods you plan to compare against. Importantly, through this section and the results section (which may include implementation details), there should be sufficient information for others to reproduce your work.
- **(25%) Results.** Describe experiments that you performed, results that you obtained, and synthesized findings and analysis. The structure of this section may vary depending on the project, but in the grading we will be looking for the thoughtfulness of your analysis. You should include graphs, tables, or other figures to illustrate your results.
- **(5%) Conclusion.** Summarize the key results, what has been learned, and avenues for future work.
- **(5%) Writing/Formatting.** Your paper should be clearly written and nicely formatted, comparable to published NeurIPS papers.
- **Contributions.** Specify the contributions of each author on the paper. This includes discussion, implementation, and writing for each part of the paper. You should also describe the contributions of any contributors not enrolled in the course (see Additional

Submission Requirements below). For an example of appropriate format, please see the author contributions for AlphaGo (Nature, 2016). However, we expect your description to include the more detailed breakdown specified here.

- **Supplementary Material.** This should be submitted as a separate file from your paper and is not counted in the 7-9 page requirement. If your project involved coding, this should include the relevant code for your project. You may also put additional visualizations, demos, videos, etc. that you wish to share with the teaching team.
  What you should not put in your supplementary material:
  - The entire TensorFlow (or PyTorch, etc.) Github source code. Only put the code you have written for the project.
  - Any code that is larger than 10 MB.
  - Model checkpoints.

Submission: You will submit your final report as a PDF and your supplementary material as a separate PDF or ZIP file. We will provide detailed submission instructions as the deadline nears.

Additional Submission Requirements: We also ask you do the following when you submit your project report:

- **Your report PDF should list all authors who have contributed to your work; enough to warrant a co-authorship position.** This includes people not enrolled in the course, such as faculty/advisors if they sponsored your work with funding or data, and significant mentors (including PhD students or postdocs who coded with you, participated in data collection, or helped draft your model on a whiteboard). All authors should be listed directly underneath the title on your PDF. Include a footnote on the first page indicating which authors are not enrolled in the course. All co-authors should have their institutional/organizational affiliation specified below the title, and their role should be described in the Contributions section of the paper.
- **Any code that was used as a base for projects must be referenced and cited in the body of the paper.** This includes course assignment code, and fine-tuning example code, open-source, or Github implementations. You can use a footnote or full reference/bibliography entry.
- **If you are using this project for multiple classes, submit the other class PDF as well.** Remember, it is an honor code violation to use the same final report PDF for multiple classes.

Project Final Presentation

Each student or team will present their project during the final exam slot for the course, on Monday, Dec. 9 from 3:30-6:30pm. More details will be provided closer to the date.

Project Examples

To help you get started, below are two example ideas for each type of project. Your project may be inspired by these, or you may propose your own idea. We encourage you to explore beyond the scope of the provided examples and think creatively about your project proposal.

Project 1.1: Comparative Analysis of Vision-Language Models for Radiology Question Answering

LLaVA-Med and CheXagent are two generative vision-language models that can perform visual question answering about radiology images. LLaVA-Med is a more generalist model that was trained on a very large dataset extracted from PubMed Central. CheXagent is a recent model that focuses more exclusively on radiology training data from publicly available datasets. Compare the performance of these two models for image analysis tasks that can be used to assist radiologists. Evaluate the performance of these two models for assisting radiologists in tasks such as disease classification, anomaly detection, and report generation from X-ray images. Potential datasets for your comparative analysis include CheXpert, CheXpert Plus, and MIMIC-CXR. (Hint: Reviewing existing AI literature in radiology can help identify appropriate evaluation datasets and tasks for your project.) Some types of analysis or probing that you may consider exploring include: Do the models have different biases in handling patient demographics? How do they compare in their ability to recognize overlapping symptoms? Does one model generalize better than the other? Identify misclassification, under-diagnosis, or over-diagnosis patterns for each model. Finally, propose ways to mitigate these vulnerabilities in future versions.

Project 1.2: Comparative Analysis of MedSAM, SAM, and SAM2 for Medical Image/Video Data

MedSAM, SAM, and SAM2 are foundation models for segmentation tasks, each with unique strengths. SAM, the first foundation model for segmentation, is designed for general vision tasks, while SAM2 introduces a video component, enabling segmentation over time in video data. MedSAM fine-tunes SAM on large-scale medical image-mask pairs, making it more suitable for medical applications. All three models support visual prompts, such as points or bounding boxes, to generate segmentation masks. In this project, you will conduct a comparative analysis of MedSAM, SAM, and SAM2 across a range of medical image and video data, such as CT/MRI scans and surgical videos. You may choose a relatively broad or narrow focus, depending on the depth of analysis in each domain. Possible tasks include evaluating their performance on tissue segmentation, tumor boundary segmentation, or surgical tools segmentation. Some potential data sources for this analysis include BraTS for brain tumor segmentation or public surgical video datasets like Cholec80. As you compare these models, consider questions such as: Do they differ in their ability to handle temporal/spatial changes in video or 3D data? How do they perform with low-resolution images, videos with artifacts (e.g., noise, contrast issues), or complex structures? How does SAM2's video design improve performance in video segmentation tasks? Are there specific failure cases for each model, such as under-segmentation or boundary ambiguity? Does one model generalize better across different imaging modalities (e.g., MRI, CT, and video)? Finally, reflect on the limitations of these segmentation foundation models and potential future directions for building models that are more generalizable for biomedical applications.

Project 2.1: Build an Agentic System for Pathology Diagnosis with Retrieval-Augmented Features

In class we have discussed several large pathology models, such as GigaPath, which learns strong visual representations of pathology images, and PLIP, which learns a large contrastive model from paired image and Twitter caption data. In this project, you can explore combining the strengths of these models to build a more comprehensive pathology image analysis system. This system will take a pathology image as input and output not only a classification but also a richer description of the content. First, use the pre-trained Prov-GigaPath model to identify a predicted class (e.g., cancer subtyping) for the image. Then, use PLIP to retrieve a set of similar captions corresponding to the image (this is a cross-modal retrieval, where text captions are retrieved from an existing dataset such as PLIP's textual database). Finally, combine the GigaPath prediction and the PLIP-retrieved captions, and input them into Gemini 1.5/GPT-4o to create a well-formed, useful output for a pathologist. You may also consider asking GPT to supplement the output with additional relevant literature. Evaluate different GPT prompts and compare which model produces better results.

Project 2.2: Build an Agentic System for Spatially-Aware Radiology Question-Answering

Radiology QA systems often require a detailed understanding of local information and spatial relationships in medical images, such as the location of lesions or abnormalities. For example, in response to the question, “Where is the opacity located?” a detailed answer might be, “Right of the midline, superior to the right hilum.” This project aims to combine the capabilities of MedSAM and CheXAgent to generate more detailed and accurate answers. MedSAM excels at segmenting organs and pathological regions, providing precise localization and boundary information of abnormalities. CheXAgent delivers global analysis and answers questions based on radiology images. Vision-Language Models (VLMs) and LLMs can then summarize the spatial and shape information generated by MedSAM along with the language feedback from CheXAgent, producing more detailed and context-aware answers for radiology QA tasks. One approach to implement this is to obtain the segmentation mask using MedSAM, extract the size and location of the lesions, convert them into a language description using a template, then concatenate this with CheXAgent’s response, and feed the combined input into GPT-4 mini to summarize and provide the final answer. Evaluate the performance of this agentic system on CheXbench, and consider sampling a smaller set that emphasizes spatial relationships and local details to assess its ability to handle fine-grained tasks. Key aspects to explore include the system’s ability to provide nuanced descriptions of anatomical structures and abnormalities, as well as its accuracy in answering complex queries.

Project 3.1 Multimodal Knowledge-Enhanced Thorax Diseases Classification using Chest X ray

Previous thorax disease classification models have primarily relied on visual features from medical images. In this project, you will explore how to incorporate clinical knowledge into the diagnostic process, enhancing the model’s understanding and improving classification accuracy. The goal is to combine visual inputs with language-based clinical knowledge to create a

multimodal classification system for thorax diseases. You will leverage large language models (LLMs) to parse clinical descriptions from medical knowledge bases, such as the Merck Manual of Diagnosis and Therapy, and generate descriptions that correspond to visual cues in thorax radiology images. You may utilize BioMedCLIP for zero-shot prediction, using the enriched textual descriptions to generate text embeddings. Evaluate this enhanced zero-shot model against the vanilla zero-shot setting discussed in class just using the simple template for text embedding. This enables the model to classify thorax diseases without needing explicit labels for every condition, as it can leverage richer text descriptions to infer visual features. You can also implement learnable prompt tuning (CoOp in Lecture 5) to fine-tune the model for few-shot thorax disease classification tasks, and experiment with combining the learnable prompt with the enriched textual descriptions for inference. To carry out this project, use thoracic disease datasets such as CheXpert or MIMIC-CXR, which provide paired radiology images and disease labels that can be augmented with clinical text descriptions.

Project 3.2 Instruction-Tuning Dataset Curation for Multimodal Biomedical System

The goal of this project is to curate an instruction-tuning dataset tailored for a biomedical AI system. Using surgical video understanding as an example, the task is to curate question-answer pairs by leveraging existing task-specific datasets and models for tasks such as surgical tool detection, Critical View of Safety (CVS), surgical phase recognition, and tissue recognition. This curated dataset will be used to fine-tune large vision-language models (LVLMs) to enhance their ability to generate context-aware surgical instructions and interpret complex surgical scenes. Start with publicly available datasets, such as Cholec80 for surgical phase recognition, EndoVis 2017 for surgical tool detection, Cholec80-CVS for Critical View of Safety identification, and EndoVis tissue datasets for tissue recognition. Develop templates for question-answer pairs based on surgical actions, phases, tools, and key safety moments (e.g., "Which tool is being used in this phase?" or "Has the CVS been achieved?"). Use large language models (LLMs) or vision-language models (VLMs) to automatically generate and refine these QA pairs. You can further fine-tune a pre-trained vision-language model (e.g., LLaVA-Med) using the curated dataset. The goal is to evaluate the system's ability to interpret and generate feedback for surgical video. Evaluate the model's performance using metrics such as phase recognition accuracy, tool detection F1 score, and mAP for CVS.