Enhancing Computer-Patient-Provider Interaction Analysis through Neurosymbolic Reasoning and Large Language Models

Jean Park[1], Sydney Pugh[1], Kuk Jin Jang[1], Eric Eaton[1], Debra Roter[2], Insup Lee[1], Kevin Johnson[1]
University of Pennsylvania[1], Johns Hopkins University[2]

Motivation

- Effective computer-patient-provider interaction plays a pivotal role in determining patient outcomes
- Automated pre-processing of interactions in video would facilitate sociotechnical studies

Q: How engaged is the patient and provider?

- Goal: Capture, process, and interpret the **nuances of communication** and **non-verbal cues** between patients and providers during medical consultations

Approach

- For this work, we formulate the problem as a video question answering (VQA) (e.g. User gives specific query about a video)
- Integrate Large Language Models (LLM) and neurosymbolic approaches to provide researchers and providers with tailored insights regarding specific queries

Key Components

1. LLM-based query-specfic neurosymbolic program generation
2. Library of symbolic modules which capture various aspects of video analysis, communication, etc.
3. User-feedback based parameter optimization (future work)

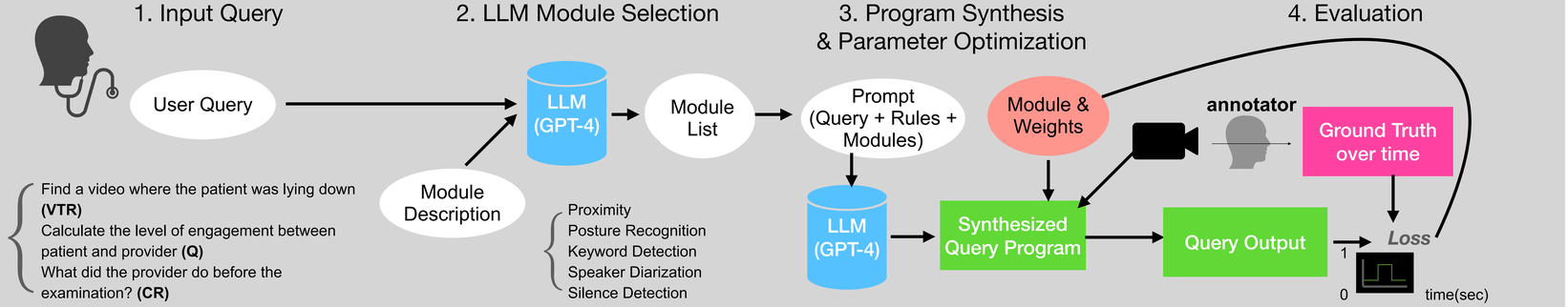
Related Works

- **Medical Video Answer Localization Task[1]**
  Locate temporal segments in medical video
- **General Video Question Answering Task[2],[3]**
  Video and audio joint encoders to answer queries
- **NeuroSymbolic Approach for VidQA[4]**
  Video information is represented by hyper-graph connecting entities and relations, and answering logic is represented by a functional program

Limitations:

1. Current VQA approaches have limitations in long-term reasoning
2. Composing multi-modal input(text, audio, video) is limited or black-box

Method



Various user query types (Video Text Retrieval, Quantification, Causal Relationship, etc) are converted to **answer localization** task

Given user query and high-level description of existing modules, GPT-4 selects relevant modules

Given video and LLM prompt, execute the synthesized query program and optimize loss by updating module weights

Loss is computed based off the intervals selected to satisfy the query and ground truth (gt)

*gt will be approximated from user feedback (future)

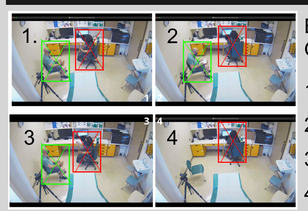
Result 1: Query for Levels of Engagement

- Query : Calculate the level of engagement a clinician will give to a patient

Output Synthesized Query Program by GPT-4

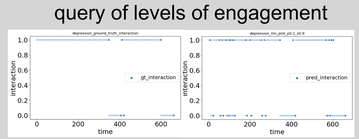
```
def calculate_engagement(weight_proximity, weight_silence, video_duration, video):
    ...
    # Calculate the level of engagement per second
    for i in range(video_duration):
        # Normalize the module outputs
        normalized_proximity = 1 - proximity[i][1] / 25 if not np.isnan(proximity[i][1]) else 0
        normalized_silence = 0 if silence[i] else 1

        # Calculate the level of engagement
        engagement[i] = weight_proximity * normalized_proximity + weight_silence * normalized_silence

    return engagement
```

Selected Modules:
'Proximity', 'Silence Detection'



Example Output
1.Talking
2.Silence
3.Talking
4.End of visit

Successful result for query of levels of engagement



Ground Truth Prediction
Minimum loss ≈ 0.15

Engagement (1) : Proximity ↓ or No silence No Engagement (0) : Proximity ↑ and Silence

Result 2: Query for Pose During Examination

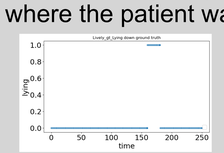
- Query : Find a video where the patient was lying down after the provider's instruction to lie back

Output Synthesized Query Program by GPT-4

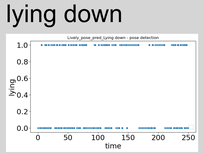
```
def find_patient_laying_down(weight, video_duration, video):
    # Define keywords that indicate the provider instructing the patient to lie back
    keywords = ['lie back', 'lay down', 'recline', 'rest']

    ...

    # Iterate over each second in the video
    for i in range(video_duration):
        # If the audio segment starts at this second and the laying pose was detected,
        #  update the relevance score
        if i in audio_segments and posture_timestamps[i]:
            relevance_scores[i] = weight['lf_keywords_segment'] * weight['lf_posture']
        # If the audio segment starts at this second but the laying pose was not detected,
        #  update the relevance score
        elif i in audio_segments and not posture_timestamps[i]:
            relevance_scores[i] = weight['lf_keywords_segment']
        # If the laying pose was detected but the audio segment does not start at this second,
        # update the relevance score
        elif not i in audio_segments and posture_timestamps[i]:
            relevance_scores[i] = weight['lf_posture']
    ...

    return relevance_scores
```

Selected Modules:
'Keyword Detection'
=> Returns segment 'after' or 'before' a keyword is detected
'Posture Recognition'
=> Returns posture

- Plot
  Less accuracy in query for finding segments where the patient was lying down

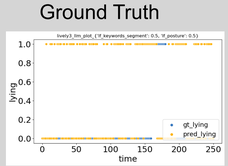


Ground Truth



Pose detection - laying

Loss ≈ 0.5
Pose detection -> acc ↓

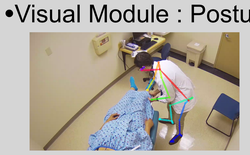


Prediction

- Audio Module : Keyword Detection (lie back)

```
(22.38, " I'm sure that it's not something that you wanna hear"),
(26.04, ' at this point in time, but given your smoking history,'),
(29.4, ' that also puts you at risk for coronary artery disease,'),
...
(77.74, ' Just gonna take a listen.'),
(90.72, ' Can I have you lie back for me?'),
(95.26, " If you're comfortable, I'll pull this up for you, too.")
...
(156.5, ' Okay, you can sit up.'), (160.52, ' Listen to your lungs.'),
(162.78, ' Take a deep breath for me.')
...
(237.82, ' Thank you.'), (238.34, ' All right, nice to meet you.'), (239.18, ' Nice to meet you also.')]
```

- Visual Module : Posture Detection (lying)



Lying down not detected

Detected as lying down

Limitation: Pose detection module does not function well for occluded and higher viewing angles

Conclusions and Future Work

- This work demonstrates the feasibility of using LLMs and neurosymbolic approaches for scalable patient-provider interaction
- Impact: By analyzing and optimizing patient-provider interactions, our systems contribute to better understanding, treatment plan compliance, and patient satisfaction

Future work:

- Integrate additional modules (e.g. object detection, sentiment analysis) for effective interaction analysis
- Develop human feedback-based optimization for answering queries
- Extend the work to augment existing Visual-Language models for general VQA tasks

References
[1] Deepak Gupta et al, "Overview of the MedVidQA 2022 Shared Task on Medical Video Question Answering", Biomedical Language Processing, 2022
[2] Zellers et al, "MERLOT: Multimodal Neural Script Knowledge Models," NeurIPS, 2021
[3] Fu et al, "An Empirical Study of End-to-End Video-Language Transformers with Masked Visual Modeling", CVPR, 2023
[4] Bo and Yu et al, "STAR: A Benchmark for Situated Reasoning in Real-World Videos", NeurIPS, 2021

Acknowledgement
This work was supported in part by ARO W911NF-20-10080

Contact info:
Jean Park - hlpark@seas.upenn.edu,
Kuk Jin Jang - jangkj@seas.upenn.edu,
Kevin Johnson - kevin.Johnson1@pennmedicine.upenn.edu