Title

Acoustic Scene Graph Reasoning: Enhancing Language Models with Structured Audio Understanding

Problem Statement

Large language models struggle to understand and reason about complex acoustic environments with multiple sound sources, their spatial relationships, and temporal dynamics. This limitation hinders their ability to perform sophisticated reasoning tasks involving audio input.

Motivation

Current approaches often treat audio as a flat sequence of features or focus on individual event detection, missing the rich structural information present in acoustic scenes. By representing acoustic environments as structured scene graphs, we can enable LLMs to perform more sophisticated reasoning about spatial and temporal relationships between sound sources. This structured representation allows for better grounding of language in acoustic contexts, potentially improving performance on audio-related tasks.

Proposed Method

We introduce Acoustic Scene Graph Reasoning (ASGR), a novel method for structuring and reasoning about complex acoustic environments. ASGR consists of three key components: 1) An acoustic scene analyzer that processes multi-channel audio to identify and localize sound sources, estimate their properties, and track their movement over time. 2) A scene graph constructor that builds a dynamic graph representation of the acoustic scene, where nodes represent sound sources and edges represent spatial and temporal relationships. 3) A graph-to-sequence encoder that converts the acoustic scene graph into a linearized token sequence suitable for LLM input. We pre-train this pipeline on a large dataset of simulated and real-world acoustic scenes paired with detailed descriptions. The LLM is then fine-tuned on a dataset of scene graphs paired with complex queries and reasoning tasks. During inference, audio input is first processed into a scene graph, then encoded into a token sequence that prefixes the LLM prompt.

Step-by-Step Experiment Plan

Step 1: Dataset Creation

Create the AcousticSceneQA dataset featuring complex, multi-source acoustic scenes paired with questions requiring spatial and temporal reasoning. Use a mix of simulated and real-world recordings. Annotate each scene with ground truth information about sound sources, their properties, and spatial relationships. Generate a diverse set of questions for each scene, covering various reasoning tasks.

Step 2: Acoustic Scene Analyzer

Implement a multi-channel audio processing pipeline using PyTorch Audio or librosa. Use pre-trained models for sound event detection (e.g., PANNs) and sound source localization (e.g., SoundLoc). Implement a tracking algorithm (e.g., Kalman filter) to follow sound sources over time. Output should be a list of sound sources with their properties (type, intensity, location) for each time frame.

Step 3: Scene Graph Constructor

Implement a graph construction algorithm that takes the output of the acoustic scene analyzer and builds a dynamic graph representation. Use networkx for graph operations. Nodes should represent sound sources, with attributes for their properties. Edges should represent spatial and temporal relationships between sources. Implement functions to update the graph over time as the acoustic scene evolves.

Step 4: Graph-to-Sequence Encoder

Implement a graph neural network (GNN) using PyTorch Geometric to process the acoustic scene graph. The GNN should output node and edge embeddings that capture the graph structure. Implement a sequence generation module (e.g., based on transformers) that takes these embeddings and produces a linearized token sequence representing the scene graph.

Step 5: Pre-training

Create a large dataset of acoustic scenes paired with detailed textual descriptions. Use this dataset to pre-train the entire ASGR pipeline (acoustic scene analyzer, scene graph constructor, and graph-to-sequence encoder) in an end-to-end manner. Use a contrastive loss to align the generated scene graph representations with their corresponding textual descriptions.

Step 6: LLM Fine-tuning

Fine-tune a pre-trained LLM (e.g., GPT-3.5 or GPT-4) on a dataset of scene graphs paired with complex queries and reasoning tasks. Use prompt engineering to format the input as: [Scene Graph Tokens] + [Question] + [Answer]. Train the model to generate the correct answer given the scene graph and question.

Step 7: Evaluation

Evaluate ASGR on the AcousticSceneQA benchmark. Compare against baselines such as: 1) Flat audio feature representation + LLM, 2) Simple event detection + LLM, 3) LLM without audio input. Use metrics including question-answering accuracy, spatial relationship prediction accuracy, and human evaluation of reasoning quality and scene understanding. Conduct ablation studies to assess the impact of each component of the ASGR pipeline.

Step 8: Analysis

Perform qualitative analysis on model outputs, visualizing the constructed scene graphs and analyzing the model's reasoning process. Identify strengths and weaknesses of the approach, and areas for potential improvement.

Test Case Examples

Baseline Prompt Input (Flat Audio Features)

Audio features: [mel spectrogram values...] Question: How many distinct sound sources are present in the scene, and what is their spatial arrangement?

Baseline Prompt Expected Output (Flat Audio Features)

Based on the audio features provided, I can detect at least 3 distinct sound sources in the scene. However, I cannot determine their exact spatial arrangement from this representation.

Proposed Prompt Input (ASGR)

Scene Graph: [Node1: car, type=vehicle, position=(x1,y1,z1), intensity=high; Node2: bird, type=animal, position=(x2,y2,z2), intensity=medium; Node3: person, type=human, position=(x3,y3,z3), intensity=low; Edge1: (car, person, distance=5m); Edge2: (bird, car, distance=10m, relative_height=above)] Question: How many distinct sound sources are present in the scene, and what is their spatial arrangement?

Proposed Prompt Expected Output (ASGR)

There are 3 distinct sound sources in the scene: 3. A person (human) Their spatial arrangement is as follows: - The car is the loudest source and is positioned at (x1,y1,z1). - The bird is of medium intensity and is located at (x2,y2,z2), 10 meters away from the car and positioned above it. - The person is the quietest source, positioned at (x3,y3,z3), 5 meters away from the car. This arrangement suggests a scene where a person is near a car on the ground, with a bird flying above them.

- A car (vehicle)
- A bird (animal)
- A person (human)
- The car is the loudest source and is positioned at (x1,y1,z1).
- The bird is of medium intensity and is located at (x2,y2,z2), 10 meters away from the car and positioned above it.
- The person is the quietest source, positioned at (x3,y3,z3), 5 meters away from the car.

explanation

The ASGR approach provides a structured representation of the acoustic scene, allowing the model to reason about the number of sound sources, their types, intensities, and spatial relationships. This enables a more detailed and accurate response compared to the baseline, which struggles to extract spatial information from flat audio features.

Fallback Plan

If the proposed ASGR method doesn't meet the success criteria, we can pursue several alternative directions. First, we could conduct a detailed error analysis to identify which components of the pipeline (acoustic scene analyzer, scene graph constructor, or graph-to-sequence encoder) are underperforming. This could lead to targeted improvements in specific modules. Second, we could explore alternative graph representations, such as hypergraphs or temporal graphs, which might better capture the complexities of acoustic scenes. Third, we could investigate the use of multi-modal transformers that can directly process both audio and text inputs, potentially eliminating the need for the intermediate graph representation. Finally, if the approach still struggles with complex reasoning tasks, we could reframe the project as an analysis paper, focusing on the challenges of integrating structured audio understanding into language models and providing insights into the limitations of current approaches.

Ranking Score: 6