Demo: A Real Time Question Answering System for Multimodal Sensors using LLMs

Xiaofan Yu
x1yu@ucsd.edu
University of California San Diego
La Jolla, California, USA

Lanxiang Hu
lah003@ucsd.edu
University of California San Diego
La Jolla, California, USA

Benjamin Reichman
bzr@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

Rushil Chandrupatla
ruchandrupatla@ucsd.edu
University of California San Diego
La Jolla, California, USA

Dylan Chu
dchu@ucsd.edu
University of California San Diego
La Jolla, California, USA

Xiyuan Zhang
xiyuanzh@ucsd.edu
University of California San Diego
La Jolla, California, USA

Larry Heck
larryheck@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

Tajana Šimunić Rosing
tajana@ucsd.edu
University of California San Diego
La Jolla, California, USA

Abstract

Question Answering (QA) establishes a natural and intuitive way for humans to interpret and understand multimodal sensor data. However, existing sensor-based QA systems are limited in the types of questions & answers, and the duration of sensor data they can handle. In this demo, we introduce an end-to-end QA system for long-term multimodal timeseries sensors powered by Large Language Models (LLMs). Our system features a novel pipeline with LLM-based question decomposition, sensor data query and LLM-based answer assembly. We further quantize the LLMs and deploy our system on two typical edge platforms, delivering higher-quality answers with low latency.

CCS Concepts

• **Computer systems organization** → **Embedded systems**; • **Computing methodologies** → **Machine learning**; *Natural language processing*.

Keywords

Question Answering, Multimodal Sensors, LLM Edge Deployments

ACM Reference Format:
Xiaofan Yu, Lanxiang Hu, Benjamin Reichman, Rushil Chandrupatla, Dylan Chu, Xiyuan Zhang, Larry Heck, and Tajana Šimunić Rosing. 2024. Demo: A Real Time Question Answering System for Multimodal Sensors using LLMs. In *The 22nd ACM Conference on Embedded Networked Sensor Systems (SENSYS '24), November 4–7, 2024, Hangzhou, China.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3666025.3699396

SENSYS '24, November 4–7, 2024, Hangzhou, China

ACM ISBN 979-8-4007-0697-4/24/11
https://doi.org/10.1145/3666025.3699396

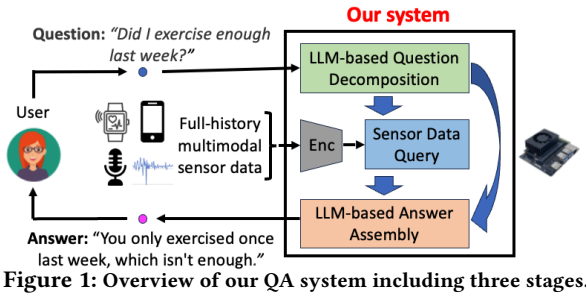

Figure 1: Overview of our QA system including three stages.

1 Introduction

In recent years, billions of sensors have been deployed across diverse applications, yet current systems struggle to provide natural user interactions. For example, answering a question like "Did I exercise enough last week?" involves complex steps, such as identifying relevant sensor data, training a machine learning algorithm to distinguish between activities, and researching health benchmarks. While activity recognition algorithms can detect specific actions [6], they fall short in addressing open-ended queries. In practice, users care more about broader patterns in sensor data and their impact on health, highlighting the need for a real-time QA system that can handle open-ended questions.

Existing QA systems for multimodal sensors are not capable of generating natural responses or handling long-term sensor data. DeepSQA [5] and AI Therapist [4] formulated the problem as a classification task, thus only allowing a limited set of answers. Recent approaches using Large Language Models (LLMs) allow more diverse questions and answers, by converting low-dimensional sensor data (such as step counts) into text [7] or using multimodal adapters [2]. However, these methods struggled with scaling to long-term time-series data from multimodal sensors.

In this demo, we present a novel end-to-end QA system as shown in Fig. 1. Our system uses LLMs on both the front- and back-end to handle diverse questions and generate natural language answers. It encodes the full history of raw timeseries sensor data into embeddings, enabling efficient searches for accurate activity information.

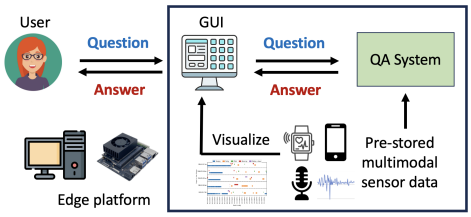

Figure 2: Demonstration setup of our system.

Our system, running on edge devices, provides a prototype solution for natural interactions between users and multimodal sensors.

2 System Overview

Our system processes user questions and full-history multimodal sensor data to generate natural language answers. It operates in three stages: LLM-based question decomposition, sensor data querying, and LLM-based answer assembly, as detailed below.

LLM-based Question Decomposition: The question decomposition stage is responsible for correctly understanding an arbitrary question and decompose it into specific query functions used in the next stage. We leverage the extensive prior knowledge in LLMs to complete such task. Specifically, we use the following prompt: "*You are a helpful assistant with querying sensor data from a database to answer user questions. Generate a query to extract the relevant sensor data using specific code functions.*" Then we instruct the LLM to highlight function names and arguments (e.g., specific activity, date to query) with different patterns. For example, "*Please highlight the function name with « »*". We use GPT-3.5-Turbo for question decomposition

Sensor Data Query: The query stage encodes large timeseries data into a compact embedding database and performs efficient searches based on query inputs. We process raw timeseries sensor data in time windows of size T. d is the number of multimodal sensors. Each sample $\mathbf{x} \in \mathbb{R}^{d \times T}$ is encoded by a pretrained multimodal encoder ϕ, and the embeddings $\phi(\mathbf{x})$ are stored in the database. We also pretrain a label encoder θ. Based on the query inputs, only the relevant data for the specified activity and time range is retrieved. For example, in response to a "how long" question, the query stage generates the text: "*You spent γ minutes doing {*`Activity`*} during {*`Time_Range`*}*", where γ is computed as follows:

$$\gamma = \sum_{t \in \texttt{Time\_Range}} [\phi(\mathbf{x}_t) \circ \theta(\texttt{Activity}) > h]. \quad (1)$$

Here h is a predefined similarity threshold.

LLM-based Answer Assembly: The final stage integrates the original question and the generated sensor context to prompt another LLM for the final answer. The prompt template is as follows: "*Answer the question based on the context below. Context: {* `Context` *} Question: {* `Question` *} Response:*" We use LLaMA2-7B quantized by AWQ [3] for answer assembly. The quantized LLaMA2 model can run locally on the edge, eliminating the need to send sensor-relevant information to the cloud and improving user privacy.

3 Demonstration

Our demonstration setup is illustrated in Fig. 2. We implement our system on two edge platforms: an edge desktop equipped with an NVIDIA 3080Ti GPU and an NVIDIA Jetson Orin NX with 16GB of RAM. Our system is accessible to users via a Graphical User Interface (GUI). Pre-collected multimodal sensor data is stored on the edge device and will be visualized through the GUI. Based on the sensor visualization, users have the opportunity to explore sensor data, compose creative questions to ask our system, and observe the generated answers in real time.

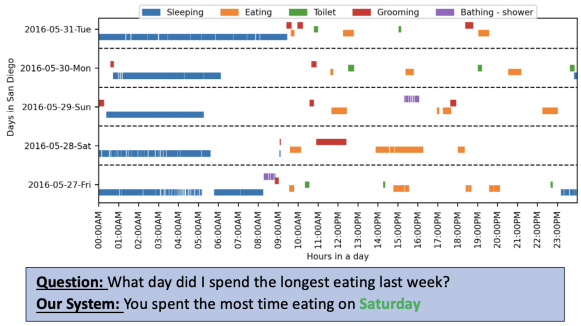

Figure 3: User experience example.

User Experience: An example user experience of our system is shown in Fig. 3. User activities detected from the sensor data are visualized using Gantt charts, where x axis represents hours in a day and y axis shows different dates. The relevant activities are shown in colored bars. Our system generates natural and accurate responses to the example questions in Fig. 3 by leveraging all three stages effectively. More specifically, in the decomposition stage, the system triggers the `calculate_duration` function with the activity "eating" and the time range "last week". During the query stage, it calculates the total eating time for each day, and in the answer assembly stage, it determines that Saturday had the longest duration based on the returned context information.

Efficiency Performance: The average latency of question decomposition, sensor data query and answer assembly stages on the edge desktop is 1.2 sec, 1.5 sec and 1.3 sec, with a total end-to-end average latency of 4.0 sec for an answer with 10-15 words. Additionally, we also deploy our system on NVIDIA Jetson Orin using the NanoLLM library [1], resulting in an end-to-end average latency of 13.4 sec. We emphasize that our system represents an initial prototype for the task. Its efficiency can be further improved by system-level optimizations such as increasing parallelism and using custom kernels.

Acknowledgments

This work was supported in part by National Science Foundation under Grants #2003279, #1826967, #2100237, #2112167, #1911095, #2112665, #2120019, #2211386 and in part by PRISM and CoCoSys, centers in JUMP 2.0, an SRC program sponsored by DARPA.

References

[1] 2024. NanoLLM. https://dusty-nv.github.io/NanoLLM/. [Online].
[2] Jiaming Han et al. 2024. Onellm: One framework to align all modalities with language. In *CVPR'24*.
[3] Ji Lin et al. 2024. AWQ: Activation-aware Weight Quantization for LLM Compression and Acceleration. In *MLSys'24*.
[4] Jingping Nie et al. 2022. Ai therapist for daily functioning assessment and intervention using smart home devices. In *SenSys'22*.
[5] Tianwei Xing et al. 2021. DeepSQA: Understanding Sensor Data via Question Answering. In *IoTDI'21*.
[6] Lilin Xu et al. 2023. MESEN: Exploit Multimodal Data to Design Unimodal Human Activity Recognition with Few Labels. In *SenSys'23*.
[7] Bufang Yang et al. 2024. DrHouse: An LLM-empowered Diagnostic Reasoning System through Harnessing Outcomes from Sensor Data and Expert Knowledge. *arXiv preprint* (2024).