ECE 535
Project Proposal
Sanskruti Khedkar, Aayan Boradia
Sept. 29, 2024

Project Topic: Home Safety System Using LLM Agents

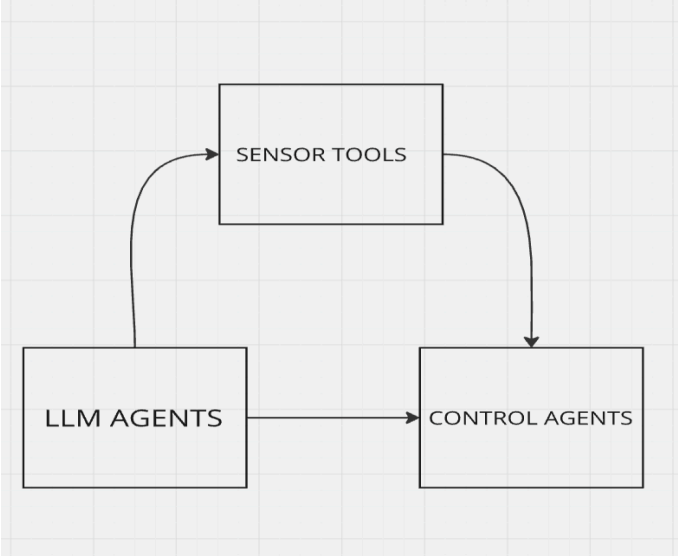
Paper for reference: https://arxiv.org/pdf/2303.17580
Langchain tutorial: https://www.deeplearning.ai/short-courses/functions-tools-agents-langchain/

Motivation: The growing need for enhanced home security has made intrusion detection a key priority in modern households. Current systems rely mostly on static rules or basic sensors, which can result in false positives or delayed responses. The goal is to create a software-based intrusion detection system that leverages the reasoning capabilities of LLM agents and publicly available data sets. This approach reduces dependency on hardware and allows easier testing and upgrades.

Design goals :
- Develop a software-based intrusion detection system using pre-recorded footage.
- Implement real-time detection using video frames or image data and object detection models.
- Utilize an LLM agent to reason about the detected events and classify potential intrusions based on behavior and context.

Deliverables :
- A working intrusion detection system that processes video footage for detecting intruders.
- LLM-driven reasoning to differentiate between normal and suspicious activity.
- Simulated alert systems such as email or SMS notifications triggered by detection.
- Final code snippets generated by the LLM for custom responses.

System blocks :



LLM Agent: Main decision-making component.

- Inputs: Raw data (e.g., video feed, sensor data).
- Outputs: Code for a safety model, and decisions for safety measures.

Sensors/Tools:

- Sensors: Cameras, smoke detectors, motion sensors.
- Tools: Fire detection model, image processing tools, alarm system activation.

Control Interface: Takes action based on LLM outputs (e.g., turn on alarms, send alerts).

SW Requirements :

- Python for programming
- OpenCV for video processing
- TensorFlow/Keras or PyTorch for object detection models
- Hugging Face for implementing LLM reasoning
- Google Colab or a local machine with a CUDA-enabled GPU for model training and execution

Team member's responsibilities: setup, software, networking, writing, research, algorithm design

Both team members will be working on everything equally yet:

Sanskruti: Research, software, writing, algorithm development

Aayan: Setup, research, software, algorithm development

Project timeline :

- October 1 - October 15: Research & Setup
- October 16 - October 31: Development
- November 1 - November 15: Testing & Refinement
- November 16 - November 30: Final Deliverables
- November 30 - December 10: Project report

References :

- A. Halimaa A. and K. Sundarakantham, "Machine Learning Based Intrusion Detection System," 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2019, pp. 916-920, doi: 10.1109/ICOEI.2019.8862784. keywords: {Support vector machines;Intrusion detection;Machine learning;Training;Conferences;Market research;Informatics;Intrusion Detection;Support Vector Machine Naive Bayes;Machine Learning},
- A. Kiran, S. W. Prakash, B. A. Kumar, Likhitha, T. Sameeratmaja and U. S. S. R. Charan, "Intrusion Detection System Using Machine Learning," 2023 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2023, pp. 1-4, doi: 10.1109/ICCCI56745.2023.10128363. keywords: {Computers;Support vector machines;Intrusion detection;Network intrusion detection;Machine learning;Software;Hardware;Support vector machine;Machine Learning;Network Intrusion Detection System;Host Intrusion Detection System;Intrusion Prevention System;Intrusion Detection System;Host;Network},