SAGE: Smart home Agent with Grounded Execution

Dmitriy Rivkin, Francois Hogan, Amal Feriani, Abhisek Konar,
Adam Sigal, Steve Liu, Greg Dudek

November 3, 2023

1 Abstract

This article introduces **SAGE** (Smart home Agent with Grounded Execution), a framework designed to maximize the flexibility of smart home assistants by replacing manually-defined inference logic with an LLM-powered autonomous agent system. SAGE integrates information about user preferences, device states, and external factors (such as weather and TV schedules) through the orchestration of a collection of tools. SAGE's capabilities include learning user preferences from natural-language utterances, interacting with devices by reading their API documentation, writing code to continuously monitor devices, and understanding natural device references. To evaluate SAGE, we develop a benchmark of 43 highly challenging smart home tasks, where SAGE successfully achieves 23 tasks, significantly outperforming existing LLM-enabled baselines (5/43).

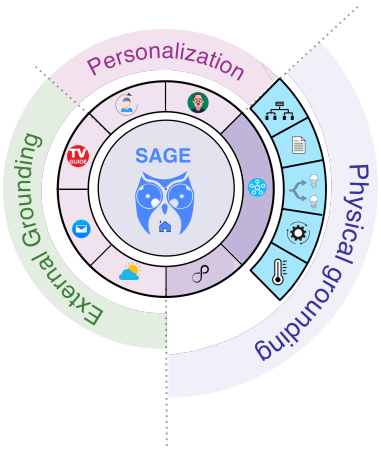

Figure 1: **SAGE - Smart home Agent with Grounded Execution.** Our system includes a collection of tools (icons described in Table 1) that can be sequentially called by the SAGE agent to accomplish a complex task. Some tools, such as the device interaction tool, follow a hierarchical structure and are themselves agents with their own set of tools, as illustrated in Figure 2.

2 Introduction

The application of LLM-based autonomous agents [1] in the smart home setting promises to revolutionize our living spaces. While voice recognition solutions have continuously improved over the last decade (Bixby, Alexa, Google Home, Siri, etc.), the planning and control of home devices have largely remained driven by the use of explicit commands such as "turn on kitchen light" and manually pre-programmed with applications such as IFTTT (If This Then That), where users specify how the system should react to a given state change through an app or online interface. These existing smart home solutions lack the flexibility to naturally interact with users, who are often required to memorize and recite specific phrases that follow a rigid syntax. This rigidity stems from the fact that the smart home planning algorithms require custom programs to be written for each use case and scenario, and are therefore unable to scale to the diverse set of needs of each user [2]. In contrast, LLM-driven autonomous agents use an LLM as the decision maker in a natural language based sequential decision making process, allowing them to react