Artificial Intelligence-based User Interaction Module for Autonomous Mobile Service Robots

Carlos Boduar Alcantara*†, Alexandre O. Júnior*, Julio Costa*, Lucas Ricken†, André Mendes*, Paulo Leitao*
* Research Centre in Digitalization and Intelligent Robotics (CeDRI), Laboratório Associado para a Sustentabilidade e Tecnologia em Regiões de Montanha (SusTEC), Instituto Politécnico de Bragança, Campus de Santa Apolónia, 5300-253 Bragança, Portugal, Email: {carlos.alcantara, alexandrejunior, juliocosta, a.chaves, pleitao}@ipb.pt
† Federal University of Technology – Parana, Via Rosalina Maria Dos Santos, 1233, Campo Mourao, Parana, Brazil

2024 IEEE 3rd Industrial Electronics Society Annual On-Line Conference (ONCON) | 979-8-3315-4031-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ONCON62778.2024.10931279

***Abstract*—With the evolution of Artificial Intelligence (AI) technologies, there is a growing demand for Intelligent Personal Assistants (IPAs) that allow fluid communication with humans. However, there are still several challenges to overcome, such as difficulty in understanding nuances in some contexts. Therefore, this study seeks to mitigate these gaps by presenting a modular and scalable architecture for integrating IPAs into autonomous service robots. The architecture allows communication between various robot systems and peripherals with AI mechanisms, including Large Language Model (LLM) and Natural Language Processing, which seek to improve the user experience. The proposed architecture was applied to developing a service robot designed to guide and interact with people in a university environment, incorporating the RASA framework with an LLM for natural language processing and response generation. The paper discusses the adopted technologies, the current state of development, the difficulties encountered, and the analysis of the first feedback from volunteers.**

Keywords: Service Robots, Artificial Intelligence, Human-Robot Interaction, Intelligent Personal Assistant

I. INTRODUCTION

Intelligent personal assistants (IPAs) are advanced computer systems driven by Artificial Intelligence (AI) techniques. They help users to perform tasks in various domains, e.g., healthcare, driving, fitness, and home management. Their popularity is mainly due to well-established virtual assistants, e.g., Google Assistant by Google, Alexa by Amazon, Siri by Apple, and Cortana by Microsoft, which enable users to employ simple voice commands to control Internet of Things (IoT) devices, add event reminders, perform online searches, and make purchases [1].

AI has never been as accessible to the public as it is today. The proof of this is the ChatGPT chatbot that reached over 100 million active users just two months after its launch, making it the fastest-growing application in history. Advanced generative models like ChatGPT are imperative milestones for AI technologies; its multimodal generative capacity allows the generation of text and image content through the understanding of human language in a conversation, opening up an infinite range of possibilities in all instantiations, including the evolution of IPAs [2].

The expansion and enhancement of IPAs contribute to the diffusion of another closely related technology: service robots. According to the International Standard Organization [3], a service robot is a *"robot that performs useful tasks for humans or equipment excluding industrial automation applications"*, a service robot can be classified into two groups: professional service robots used in inspections, visitor guidance, logistics, delivery, and medical applications, and personal service robots used for domestic tasks, entertainment, and home security [4].

Service robots work directly with humans, making the development of human-robot interaction (HRI) skills essential. The first step to address this challenge is to implement responsive "social interfaces," which serve as a focal point for interaction by providing a stimulating and user-friendly experience [5]. The second step involves establishing effective communication between the robot and the human. With the emergence of Large Language Models (LLMs), robots can now understand and interpret human language and generate concrete responses [6]. Henceforth, this advancement allows the service robots to function as advanced mobile IPAs.

However, the avant-garde incorporation of LLMs into service robots still presents adversities. Language models struggle to understand contextual nuances, which leads to misinterpretations in user conversations. These misconceptions often lead to frustrating interactions and hinder the efficient management of the robot's functions. To overcome these problems, a more streamlined integration of the robot's control modules and peripherals with the embedded IPA system is needed. Additionally, it is essential to establish a robust tracking system for the user conversation history to better control robot actions and behaviour according to the dialogue context, implying finding solutions to generalize and make language models versatile enough to respond to unforeseen scenarios, thus trying to overcome the scalability limitations of these models [7].

This paper introduces a scalable and modular architecture to address the existing gaps in the effective integration of IPAs into service robots. This architecture connects a service robot's various systems and peripherals with advanced AI mechanisms, enabling the deployment of an iterative chatbot module that enhances conversations through natural language generation and improves user experience. The proposed architecture has been applied to developing a service robot for guiding and iterating with people in a university setting. The preliminary results will be discussed, including the technologies adopted, difficulties, insights concerning the first users' feedback, and the next steps in its implementation.

The rest of the paper is organized as follows: Section II provides a brief perspective on dialogue-driven HRI aspects in interactive service robots. Section III defines an architecture for an AI-supported interactive autonomous service robot, covering different integration layers with modules for interfacing, chatbot, navigation, and information requests to APIs and remote servers. Section IV describes the developed visitor guide service robot in a university environment. Section V discuss the preliminary results of its implementation, highlighting the challenges and lessons learned from users' reviews. Finally, Section VI summarizes the main findings and points out the future work.

II. USER INTERACTION WITH SERVICE ROBOTS

Mobile services and social robots are becoming increasingly prevalent in various facilities. These robots are designed to navigate autonomously and facilitate HRI through AI conversational modules. An example is the Lio-A personal robot assistant [8], specifically designed to interact with patients and staff in medical environments. Based on the Robot Operating System (ROS) middleware, its control system allows it to navigate autonomously to transport samples and carry out routines, handle objects with a collaborative robotic arm, and communicate with people using AI algorithms for facial identification, voice recognition and speech synthesis.

In [9], a service robot was designed to act as a museum tour guide, accompanying visitors through the galleries, presenting works of art, and answering questions. For this purpose, an ROS-based localization system and a dialogue framework supported by Google Dialogflow were developed. While this cloud-based application facilitates the development of IPAs, one of its primary constraints is the need for a stable internet connection, which may not always be available as the autonomous robot moves through the environment.

Some dialogue systems, such as the open-source RASA platform, can be wholly embedded in robotic platforms to create and manage customized chatbots capable of communicating with users in different conversational scenarios [10] and bypassing the problems generated by network connection failure or latency. RASA supports the natural language understanding (NLU) functionalities, interpreting the user inputs through rule-based policies and machine learning (ML) techniques to determine system actions according to the conversation history and context [11].

RASA also includes functionalities for extracting and classifying intentions (sets of actions and general purpose when given, for example, commands or asking questions) and entities (terms or phrases within the user's message that contain information or details necessary for context awareness). Both resources are indispensable to the chatbot's dialogue systems since the conversation structure is constituted chiefly and managed by rule-based algorithms hand-crafted by the developers, who use intentions and entities to construct the conversation flow according to the chatbot's specific application, creating personalized responses and automating tasks to generate a coherent conversation. Nevertheless, this is one of the most significant limitations of this type of chatbot, as developing rule-based scripts is time-consuming and restricts the conversation to a very limited set of scenarios, which can be frustrating for the user [12].

LLMs can serve as complementary units to the robot dialogue system, flexibly processing a wide range of user communication variation scenarios, offering a personalized user interaction experience, better guiding the triggering of functionalities embedded in the robot, and streamlining the IPA model training process. Nevertheless, LLMs can introduce errors and risks of hallucination, causing the robot to assume unexpected behaviors or inconsistent statements [13], making it essential to ensure the synergistic communication between the LLM and the dialogue model.

In this perspective, this paper presents an architecture for service robots, which incorporates a chatbot system and LLM to guide the robot's interaction with the user, based on the robot's implemented functionalities and the context of its application, taking advantage of the chatbot, while simultaneously opening up the possibility of the robot's engagement in more complex conversations in diverse scenarios with LLM.

III. SERVICE ROBOT ARCHITECTURE

Specifying the architecture of an autonomous service robot is a highly complex task due to the need for the seamless integration of various modules and components, including sensors, control and navigation systems, and user interface elements. Besides, ensuring the system's scalability is crucial, allowing for the incorporation and development of new modules over time. Figure 1 illustrates the proposed architecture for an autonomous service robot designed to leverage AI-supported virtual assistant modules for enhanced robot-user interaction integrated with the robot's interface and control systems and provides an expandable layer for actions and functionalities.

The *Robot Interface* layer comprises elements responsible for interacting with the user and the environment within the service robot's operational domain. The robot's sensing module, equipped with sensors, enables data collection from the environment for localization and navigation. The camera initiates the user approach when detecting a person. The user can communicate with the robot via the interactive interface with selection menus or by the microphone. The interface also includes a speaker for the robot's responses.

The main AI-supported mechanisms are in the *Virtual Assistant Chatbot* layer. The user's audio input is processed through a speech-to-text converter algorithm before being sent to the natural language interpreter (NLI), similar to the text inputs from the interactive interface's selection menus. The NLI is responsible for processing natural human language and converting this information into a format that the computer can understand. Following this conversion, the text undergoes an NLU process, where the structure of the inputs is analysed to classify the user's intentions and entities.

The chatbot core is the virtual assistant system's backbone. It is responsible for managing the conversation, determining how the virtual assistant should respond, and deciding on the following actions. The chatbot core uses ML to make

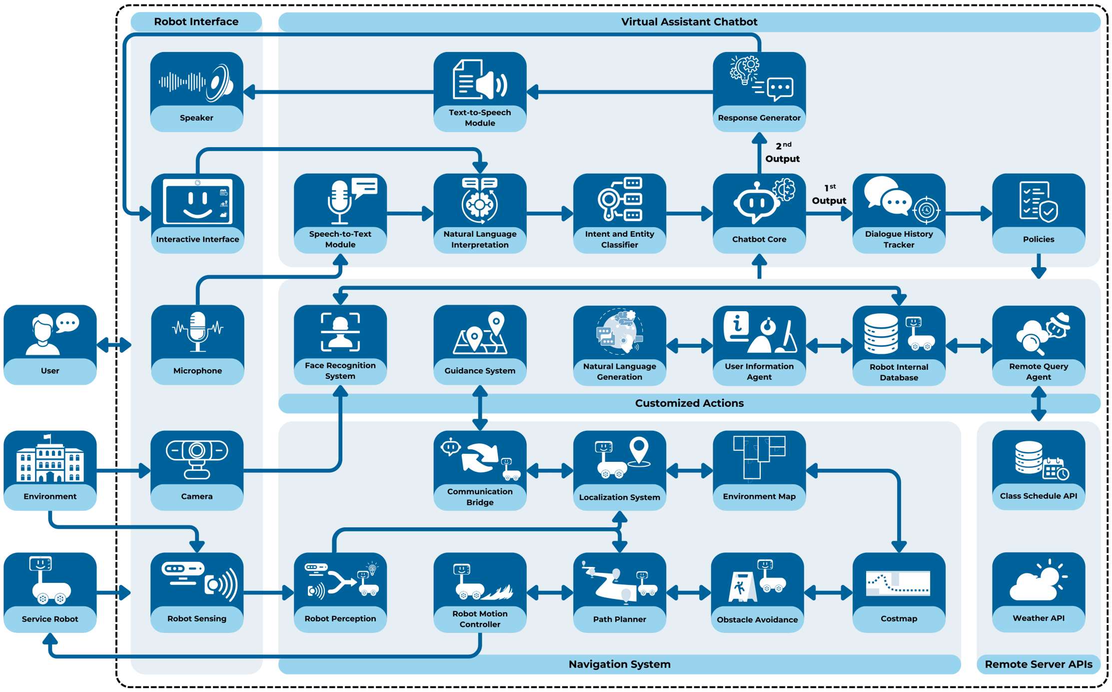

Figure 1. Modular architecture for an autonomous service robot with AI-based interaction system.

decisions based on the intentions and entities extracted, and the dialogue history tracker predicts the best action sequence based on the conversation's current state and policies. Policies are pre-defined rules used to train the system, which defines how the assistant should act in different situations, e.g. whether it should return an answer or question to the user, ensuring that the answers are coherent and contextually appropriate, or whether it should call an external function or action that performs a specific operation.

Once the action has been carried out, the chatbot core triggers the response generator, which formats the reply to be sent based on a defined domain as a response template or based on the customized action that has been carried out, which can enrich the responses based on complex logic or queries to external APIs, for example. The user can visualize the response generated via a caption on the interactive interface, or it can be passed through a text-to-speech module so that the user can hear it through the interface's speaker.

Customized Actions represent a scalable layer of customized modules capable of performing different functions. For instance, the facial recognition system enables the robot to recognize a user from its internal database or to add a new one and include its name in the conversation. The user information agent manages the data needed to form the response according to the dialogue history and policies, retrieves existing information from the robot's internal database, or triggers the remote query queue agent if necessary. This agent makes requests on the *Remote Server APIs* layer, where different services can provide information that does not exist in the internal database. In this work, as the service robot is proposed in a university guiding scenario, the APIs exemplified in the architecture are used to obtain the schedule of classes, teachers and university events, as well as local weather information. However, the choice of APIs can be extended and adapted to any robotic platform's purpose.

The user requests can often deviate from the policies in the chatbot's trained model or require information that does not exist in the robot's memory. The architecture proposes a solution where the user information agent can request a natural language generator, such as an LLM. These models are trained using non-supervised ML techniques with vast amounts of unlabeled text parameters that allow the creation of content and responses, thus maintaining a dialogue line with the user. Another proposed module in the actions layer is the guidance system, where the service robot can provide directions to a specific location, such as a laboratory or classroom, or guide and accompany the person to the location.

The guidance system connects to the *Navigation System* layer via a communication bridge that translates the requested destinations to the robot's low-level localization and control system. The perception module processes raw data from the robot's sensors (e.g. lidar, odometry and cameras) and extracts the information needed by the path planning and localization systems, the latter of which can take advantage of sensory fusion and methods such as Simultaneous Localization and Mapping (SLAM), Kalman or particle filters to accurately determine the position and orientation occupied by the robot using a map of the environment as a reference.

The path planner determine the optimum route for the robot to reach its destination. The obstacle avoidance system, adjusts the robot's path in real-time to avoid collisions. Costmaps help in this process by being derivations of the environment map that determine the ease or difficulty of moving through a given location and help perceive variations in the environment. The motion controller receives trajectory commands and generates the necessary signals for the robot's actuators to execute the required trajectory using PID control algorithms.

IV. IMPLEMENTATION OF A VISITOR GUIDANCE ROBOT

The modular architecture described in Section III is being implemented to develop an autonomous mobile service robot. That guides and interacts with students, staff, and visitors in a university environment. Figure 2 depicts the preliminary version of its implementation, pointing out to the main hardware components and the present elements in the interactive interface.

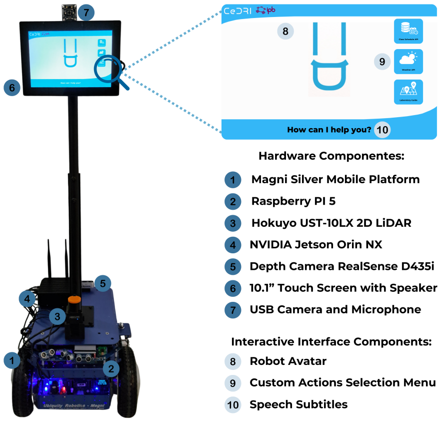

Figure 2. Preliminary version of the visitor guidance robot.

The Magni Silver robot from Ubiquity Robotics is a modular mobile platform compatible with ROS. It simplifies the integration of robotics tools and libraries and the addition of external sensors and hardware components. Two embedded systems have been deployed to optimize the service robot's data processing: the Raspberry Pi 5 for the robot's navigation and the Jetson Orin NX for the IA-based interaction modules.

The chatbot's virtual assistant layer was developed using the RASA framework. A foundational dialogue structure was designed to conduct the user interactions with the chatbot in a semi-deterministic manner, seamlessly integrating the service robot's diverse functionalities while striving to maintain user engagement and avoid monotony. Figure 3 provides an exemplification of the chatbot's conversation flow.

The user inputs are processed by the RASA's NLU component, which interprets the messages by identifying and extracting intents and entities. The NLU was trained with sets of phrases and their respective elements designed to reflect typical user responses according to the pre-defined interaction scenarios. If the intentions have been correctly recognized, the dialogue management system, also known as RASA Core, determines the following action to be taken in accordance to the defined policies and the dialogue history tracker. Consequently, the RASA Core generates responses according to the conversation's current state concerning the narrative pipeline.

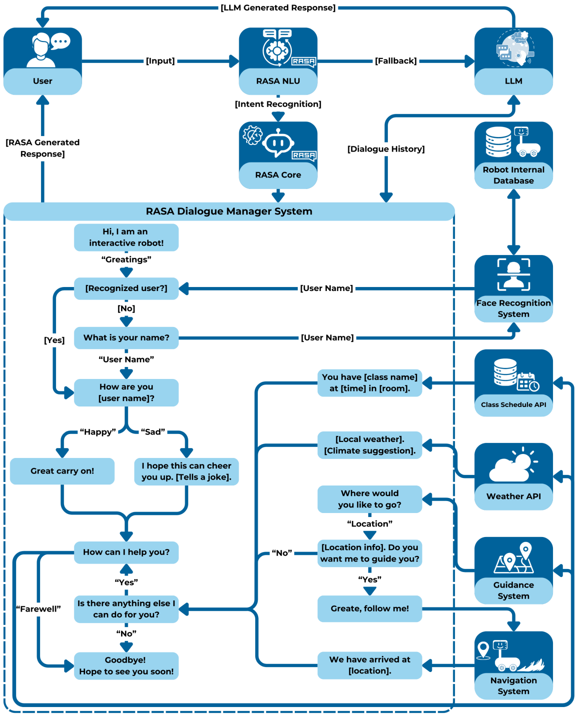

Figure 3. Exemplification of the service robot's dialogue system.

As part of the conversation flowchart, the robot starts by greeting the user, if recognized by the facial recognition module, the robot directly asks how he is. Otherwise, it asks for the person's name and saves this information in its database. This module, along with other functionalities, has been integrated into the chatbot through the RASA custom actions feature. This feature allows external functions and services to be called based on patterns identified by RASA or patterns defined with pre-established intents.

To foster empathy and deepen the conversation, the chatbot asks how the user is feeling. If the response indicates a negative emotion, such as sadness, attempts to lift the user's enthusiasm with jokes. Following this, the chatbot offers assistance by inquiring how it can help. Among the available features is the weather API, which provides the local weather forecast and offers recommendations based on it, as well as the class schedule API, which displays lesson times and locations, and university events. Both features are managed by the user information module, which, through the remote query agent, periodically makes requests to remote servers to retrieve information, such as the university website.

The main functionality of the robot is the guidance module. When selected, the robot provides information about the user's desired destination and, if requested, can accompany the user to the location. The communication bridge module was implemented to establish the communication between the RASA chatbot and the navigation layer developed in ROS. This module employs serial communication between the Raspberry Pi and the Jetson, publishing received destinations to a ROS topic and notifying the chatbot when the robot reaches its destination, allowing the conversation flow to proceed seamlessly. Once the destination is published to the topic, a Dijkstra algorithm is used to plan the global path for the robot's displacement from its current position.

The service robot's navigation layer was developed using the ROS *Navigation Stack* library, which leverages the robot's sensors, perception and environment maps to generate local costmaps to enable route planning, obstacle avoidance, and motion control. Initial testing was conducted in a simulated environment [14] before deployment in the university setting. The environment map was created using the SLAM-bas *Gmapping* library, which leveraged the robot's odometry data and Hokuyo 2D LiDAR readings to generate a grid occupancy map. This map, combined with the robot's sensor data, serves as input for the implemented localization system based on the Adaptive Monte Carlo Localization (AMCL) particle filter.

The university setting is highly dynamic due to the intense flow of people and features areas with few distinctive characteristics, such as long homogeneous corridors. For this reason, the AMCL filter was integrated with a fiducial marker detection algorithm supported by the *ar_track_alvar* library to mitigate intrinsic localization errors. Several fiducial markers were placed on the ceiling of the environment, leveraging the data from the RealSense camera, the system determines the relative position between the markers and the robot, enabling corrections to the AMCL position estimation.

As previously discussed, the NLI of chatbots, such as RASA, is constantly subject to failure since it is only possible to predict some user behaviors and input scenarios. To address this problem, when the RASA NLU experiences a fallback in the intent classification, the custom action of the user information agent is triggered, forwarding the user's incoming message along with the conversation history to the LLM incorporated into the robot's chatbot, the TinyLlama-1.1B model [15]. With 1.1 billion parameters, TinyLlama seeks to balance performance with efficiency in terms of computational resources, making it suitable for deployment in embedded systems, and even though it is much smaller than LLMs such as GPT-4, it is still powerful enough to perform a wide variety of natural language processing applications. Due to its small size, specific information tends to be incorrect, so in order to be able to answer the desired information adequately, the model was fine-tuned with data related to the institution.

Therefore, using the RASA framework to develop the service robot's IPA, the robot can understand the user input in typical conversations, generate appropriate responses that align with the developed story, and track and store the conversation history. At the same time, the TinyLlama can access this history and the user's input when it bypasses the RASA-trained model, which allows it to proceed with the conversation, interpret more complex inputs, and generate more fluid and information-rich conversations and responses, enhancing the user experience. When the RASA NLU succeeds in extracting the intentions, the conversation returns to the standard story.

Lastly, an HTML application was developed for the service robot's interactive interface. The avatar features simple animations, enabling it to blink, talk, and smile. The interface includes menus for selecting customized actions, such as the guidance system, and accessing weather and class schedule APIs, which help users familiarize themselves with the robot's functionalities. Additionally, a subtitle system is incorporated to allow users to read the robot's speech output.

V. DISCUSSION AND LEARNING LESSONS

Based on adaptations of the work [16], a concept test was carried out with 15 people to find out how they felt about interacting with the robot and various aspects of integration between the systems. The scores ranged from 1 (strongly disagree with the statement) to 5 (strongly agree with the statement), and the results are shown in Table I. The participants generally had a good experience interacting with the robot without any significant difficulties. However, as shown in statement 6, there were inconsistencies in the answers obtained due to the small training dataset of the Rasa model, causing the system to take inappropriate actions, and the small dataset used in fine-tuning, causing the LLM to hallucinate and thus providing inaccurate answers.

Table I

RESPONSES TO INDIVIDUAL SUS STATEMENTS.

N°	Statement	Avg.	Var.
1	I think that I would like to interact more with the robot.	4.27	1.21
2	I found the interface unnecessarily complex.	1.40	0.40
3	I thought the interface was easy to use.	4.47	0.55
4	I think that I would need the support of a technical person to be able to interact with the robot.	1.67	1.09
5	I found the various functions of the robot were well integrated.	4.40	0.54
6	I thought there was too much inconsistency in the robot's responses.	2.80	1.02
7	I imagine that most people would learn to interact with the robot very quickly.	4.33	0.95
8	I found the interface not user-friendly.	1.47	0.27
9	I felt very confident in interacting with the robot.	4.47	0.41
10	I needed to learn a lot of things before interacting with the robot.	1.33	0.67

It is necessary to improve the datasets used, increase them, and give specific information, such as relevant projects developed by the university's research centres or details about its teaching staff. This improvement enables better management of the LLM usage and effective sharing of conversation history, as well as prevents generated conversations from deviating too far from the robot's operational history, ensuring that the conversation can return to RASA and even avoiding LLM hallucinations. Other improvements include features such as small talks during the guiding process to maintain an interactive dialogue with the user. To achieve this, the remote query agent can retrieve recent news and interesting facts, such as sports news, movies playing in theatres, or popular and local cultural events. Incorporating this information into the conversation can make interactions more engaging and help users empathize with the robot.

Many improvements are still needed to achieve a genuinely social interface. Developing a more comprehensive range of animations is essential, allowing the robot to react to people's feelings according to the context of the conversation. Additionally, giving the robot its own personality is crucial for creating a more engaging and relatable interaction. Such enhancements would help the robot to build a more personal and empathetic connection with users, making interactions feel more natural and human-like. Moreover, the robot's body must still be deployed to make it more eye-catching and protect the sensors and chassis components. A 3D model is being developed and will be used to create a resin mould. This will result in a lightweight and durable structure that will not impact the robot's battery performance.

The Magni Silver mobile platform has proven to be versatile for integrating hardware modules and the navigation layer, due to ROS compatibility. The main downside is that the control drivers are currently only available for ROS Noetic, with no ROS 2 version. The use of a Raspberry Pi 5 enhances the processing power, improving the localization system performance; however, this required a dockerization approach for ROS Noetic, as the Pi 5 architecture does not support Linux versions compatible with Noetic, like Ubuntu 20.04. This solution in a containerized environment, ensuring compatibility and proper operation of the robotic platform.

The speech-to-text and text-to-speech modules are the primary focus of current service robot development efforts. As previously mentioned, the modules most critical to the robot's operation must be implemented in an embedded manner to avoid malfunctions if the robot loses connection during navigation. The university environment is extremely noisy, complicating speech recognition. Preliminary tests with *SpeechRecognition* module were unsatisfactory, being necessary to test other libraries. The *SpeechSynthesis* tool is currently used for speech generation; in its current state, this module was implemented using the Google Text-to-Speech (gTTS) library, but it is necessary to explore other tools to create a more user-friendly voice for the interface. Additionally, the noise-cancelling microphone arrays will be tested to improve the voice recognition performance and to develop the robot's approach system, enabling it to address the person providing the voice to initiate the dialogue.

VI. Conclusions and Future Work

Developing AI-based user interaction modules for service robots marks a milestone in advancing HRI. It paves the way for the effective use of IPA on mobile platforms. This work proposes a scalable, modular architecture for service robots to integrate various intelligent user interaction modules with the robot's control functionalities and peripherals. A guidance robot combining advanced AI mechanisms such as the RASA dialogue management system and the TinyLlama LLM is being developed to validate this architecture. The integration of these mechanisms enhances the user experience by improving the interaction and conversation scenarios, thereby directing the platform's functionalities through natural dialogue.

Future work will focus on developing robust speech-to-text and text-to-speech modules using different libraries, enhancing the robot's empathetic capabilities, and refining its interaction mechanisms to ensure more natural communication. Improvements in the user information agent and fine-tuning the LLM with specific datasets will further strengthen the robot's ability to provide accurate and contextually relevant responses. Key intentions and entities of the chatbot's conversation history will be extracted, and what-if simulations will be conducted to generate response scenarios. These scenarios, integrated with reinforcement learning approaches, will help to improve the platform's conversational performance. Additionally, expanding the range of animations and developing a unique personality for the robot will foster stronger emotional connections with users.

Acknowledgments

This work was supported by national funds through FCT/MCTES (PIDDAC): CeDRI, UIDB/05757/2020 (DOI: 10.54499/UIDB/05757/2020) and UIDP/05757/2020 (DOI: 10.54499/UIDP/05757/2020); and SusTEC, LA/P/0007/2020 (DOI: 10.54499/LA/P/0007/2020).

References

[1] E. Islas-Cota, J. O. Gutierrez-Garcia, C. O. Acosta, and L.-F. Rodríguez, “A systematic review of intelligent assistants,” *Future Generation Computer Systems*, vol. 128, pp. 45–62, 2022.
[2] T. Wu *et al.*, “A brief overview of ChatGPT: The history, status quo and potential future development,” *IEEE/CAA Journal of Automatica Sinica*, vol. 10, no. 5, pp. 1122–1136, 2023.
[3] *Robots and robotic devices – Vocabulary*, International Organization for Standardization Std. ISO 8373:2012, 2012.
[4] J. A. Gonzalez-Aguirre *et al.*, “Service robots: Trends and technology,” *Applied Sciences*, vol. 11, no. 22, p. 10702, 2021.
[5] C. McGinn, “Why do robots need a head? the role of social interfaces on service robots,” *International Journal of Social Robotics*, vol. 12, no. 1, pp. 281–295, 2020.
[6] T. Sievers and N. Russwinkel, “Interacting with a sentimental robot – making human emotions tangible for a social robot via ChatGPT*,” in *IEEE International Conference on Advanced Robotics and Its Social Impacts (ARSO)*, 2024, pp. 182–187.
[7] C. Zhang *et al.*, “Large language models for human-robot interaction: A review,” *Biomimetic Intelligence and Robotics*, p. 100131, 2023.
[8] J. Mišeikis *et al.*, “Lio-a personal robot assistant for human-robot interaction and care applications,” *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5339–5346, 2020.
[9] S. Rosa *et al.*, “Tour guide robot: a 5G-enabled robot museum guide,” *Frontiers in Robotics and AI*, vol. 10, 2024.
[10] A. Fujii and J. Kristiina, “Open source system integration towards natural interaction with robots,” in *17th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2022, pp. 768–772.
[11] T. Bocklisch, J. Faulkner, N. Pawlowski, and A. Nichol, “Rasa: Open source language understanding and dialogue management,” *ArXiv*, vol. abs/1712.05181, 2017.
[12] K. Jokinen, “Conversational agents and robot interaction,” in *International Conference on Human-Computer Interaction*. Springer, 2022, pp. 280–292.
[13] C. Y. Kim, C. P. Lee, and B. Mutlu, “Understanding large-language model (llm)-powered human-robot interaction,” in *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, 2024, pp. 371–380.
[14] A. de Oliveira Júnior, L. Piardi, E. G. Bertogna, and P. Leitão, “Improving the mobile robots indoor localization system by combining slam with fiducial markers,” in *Latin American Robotics Symposium (LARS)*, 2021, pp. 234–239.
[15] P. Zhang, G. Zeng, T. Wang, and W. Lu, “Tinyllama: An open-source small language model,” 2024.
[16] A. Bangor, P. T. Kortum, and J. T. Miller, “An empirical evaluation of the system usability scale,” *Intl. Journal of Human–Computer Interaction*, vol. 24, no. 6, pp. 574–594, 2008.