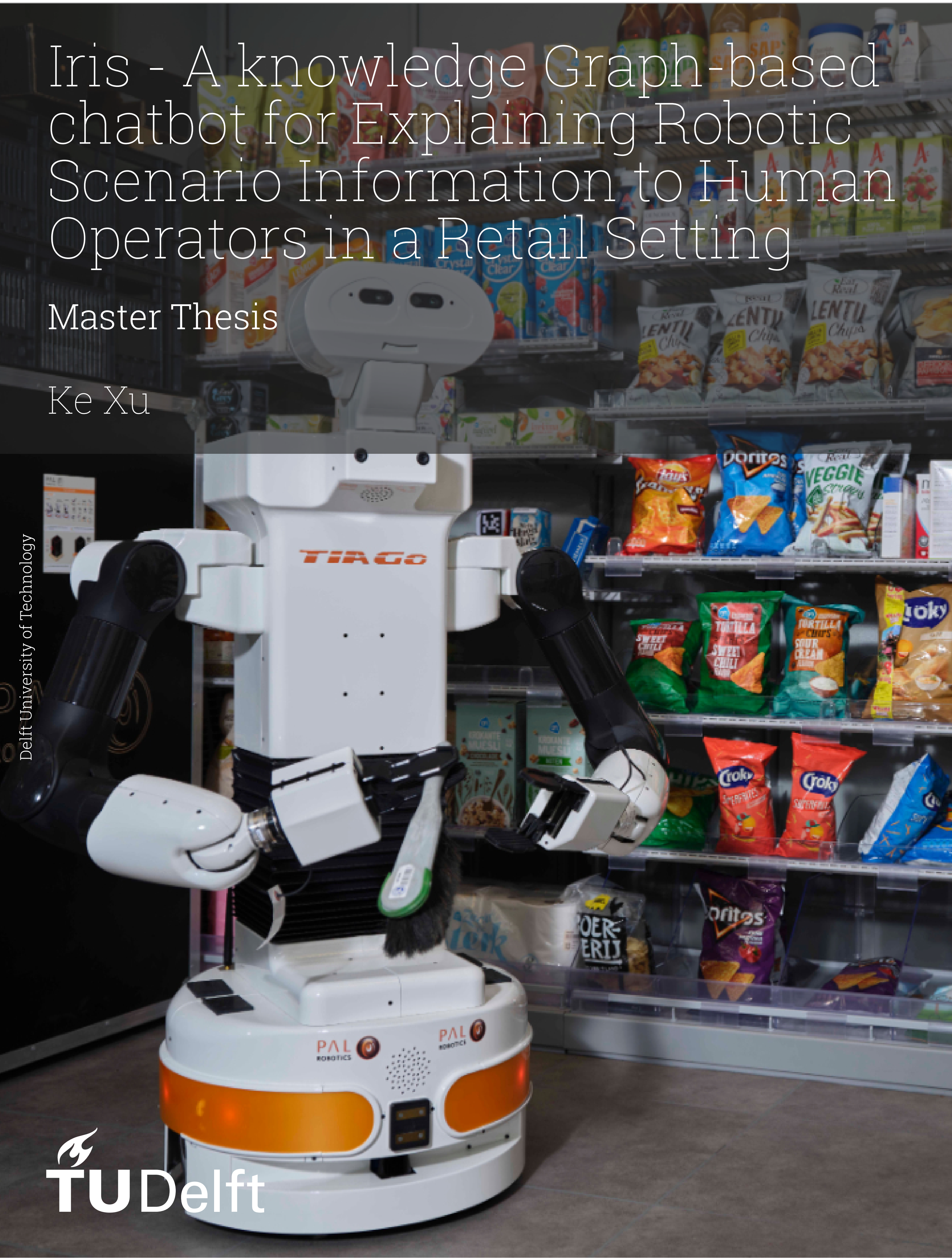
Iris - A knowledge Graph-based chatbot for Explaining Robotic Scenario Information to Human Operators in a Retail Setting

Master Thesis

Ke Xu



Delft University of Technology

TU Delft

Iris - A knowledge Graph-based chatbot for Explaining Robotic Scenario Information to Human Operators in a Retail Setting

Master Thesis

by

Ke Xu

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Tuesday March 21, 2023 at 10:00 AM.

Student number:	5290864	
Thesis committee:	Dr. Carlos Hernández Corbato,	TU Delft, supervisor, chair
	Dr. Yke Bauke Eisma,	TU Delft, committee member
	PhD candidate Corrado Pezzato,	TU Delft, committee member

TU Delft

Acknowledgement

With this thesis, I will obtain my master's degree in Robotics at Delft University of Technology. I would like to extend my deep gratitude to all those who have offered me a lot of help and support in the whole process of my thesis project. This valuable learning and reflection opportunity will act as a starting point in my future study.

First and foremost, my sincere thanks go to Dr. Carlos Hernández Corbato, my supervisor, who always offered valuable comments and suggestions in our meetings with patience and understanding. With his expertise and supervision during my study, I could accomplish this project.

Also, I owe thanks to my colleague Yuan Sen, who helped and encouraged me throughout the thesis writing. Besides, I appreciate all participants involved in my experiment and the continuous support from people in KAS Group.

Last but not least, I sincerely express my gratitude to my family and friends, who always take care of my anxiety, frustrations and happiness.

Ke Xu
Delft, March 2023

Iris - A knowledge Graph-based chatbot for Explaining Robotic Scenario Information to Human Operators in a Retail Setting

K. Xu, S. Yuan and Carlos Hernández Corbato

***Abstract*—The problem of assisting users in comprehending the robotic scenario information in a retail setting has been studied. To design the system, an integrated ontology composed of several IEEE standard ontologies and a labelled property graph (LPG)-based ontology modified from the Web Ontology Language (OWL)-based ontology was proposed to symbolize information in the robotic environment. Then, a knowledge graph (KG)-based chatbot was developed to provide natural language interaction with users. A case study in a retail setting was designed, and the results were analyzed. The effectiveness of our designed system has been experimentally validated in both static and dynamic scenarios, with at least 1.5 times improvements.**

***Index Terms*—Ontology, Knowledge Representation, Knowledge Graph, Chatbot, Human-Robot Interaction, Rasa**

I. INTRODUCTION

WITH the advent of robots and intelligent agents, their intelligent behaviours have been explored to provide convenience in many domains. As the agent functions as a mediator in consequential decision-making, the agent's explainability is essential for the end-user to take informed, accountable actions [1]. The purpose of explanations is either to help users gain confidence or establish trust towards the system has been suggested in [2]. Most research focuses on algorithmic transparency as a method of overcoming the opaqueness of black-box algorithms [3]. However, other studies explored the effectiveness of direct verbal or non-verbal explanations on robot behaviours towards human trust and understanding [4], [5], [6], [7]. These studies provided systems with transparency by facilitating user understanding of the agent's intent, behaviour, plans and decision-making process [8]. Inspired by their work, the system's explainability has been extended to real-time robotics scenario knowledge containing static environment and dynamic task information in this work.

A knowledge graph-based chabot named Iris to ground our idea by interpreting robotic scenario information for users has been presented. The framework includes a ROS environment (RE) module to gather robotic task data generated in ROS stimulation; a knowledge graph (KG) module for storing the knowledge of robotic scenario data in a constructed database management system (DBMS); and a conversational agent (CA) module that retrieves corresponding information based on recognised entities and intentions from user utterances to form responses. Furthermore, a case study is adopted to examine the effectiveness of the designed chatbot in a retail setting.

The main contributions of the paper are listed as follows:

1) This paper proposed an integrated ontology composed of several IEEE standard ontologies and modified the Web Ontology Language (OWL)-based ontology [9] to labelled property graph (LPG)-based ontology [10], [11], making it useful for symbolizing robotic environment knowledge.
2) A KG-based chatbot was proposed to solve the difficult comprehension between the users and robotic scenario information.
3) A case study in a retail setting is designed to evaluate the performance and effectiveness of the system.

The rest of the article is organized as follows. In section II, some related work about knowledge representation in human-robot interaction (HRI) and conversational interaction in HRI are provided. The proposed methods are demonstrated in section IV. The designed experiment and the evaluation results are summarised in section V. Finally, conclusions are drawn in section VI.

II. RELATED WORK

Human-robot interaction (HRI) has become the key component in current robot applications. Its applications can be divided roughly into three categories if only robots under the narrow definition are considered: 1) human supervisory control of robots, 2) telerobotics in specific domains and 3) human-robot social interaction [12]. With the spreading use of robots in daily human life, research on the third interaction mode has emerged in recent years [13]. Among various information exchange approaches in human-robot social interaction, spoken language interaction has the potential to be the most natural and efficient way [14].

Hence, we consider conversational interaction as a solution to implement the explainability of robotic scenario information. Motivated by this goal, the system must first have a representation of the environment's knowledge. Then, natural language processing (NLP) technology is used to complete the interaction progress and assist users in comprehending scenario-related information. The following sections summarise related research on knowledge representation (KR) in HRI and conversational interaction in HRI.

A. Knowledge Representation in HRI

A semantic model based on first-order logic was proposed in [15], which only suits a simple scene containing limited objects. A task ontology for an elderly-care dialogue system with the help of caretakers was adopted in [16], where the ontologies are task-oriented with a structured hierarchy to indicate tasks and subtasks for the system to execute. To overcome the problem of limited conversational topics, an extended method for the knowledge base (KB) in run-time dialogue interaction was described in [17]. The method followed an

OWL2 ontology framework based on description logic [18] to encode concepts in TBox at a generic, culture-agnostic level and instances in Culture-Specific ABox and Person-Specific ABox to adapt the cultural background of users. It is worth noting that knowledge representation is used in these studies as a way of inferring robot task execution, which is contrary to our purpose of storing environmental knowledge for access by the conversational agent.

Compared with the above methods, KG proposed by Google [19] emphasises the relationship between data, which represent entities and relationships graphically and intuitively [20]. Besides, it includes information about specific individuals based on its schema-ontology that models only general types of things with certain properties [21]. Due to these modelling advantages, it was chosen as the KR method to construct a human-readable KB for the interactive assistant in this project.

B. Conversational interaction in HRI

The Dialogflow platform and its supportive Natural Language API were used to capture the intent and entities from users' utterances [17]. In addition to using the development platform directly, some natural language understanding (NLU) algorithms were implemented to enable conversational interaction. Deep semantic role labelling (SRL) [22] was adopted to determine the predicate-argument structure in users' instructions to command robotic tasks for industrial robots [23]. Conditional random field (CRF) [24] was utilized to extract task-related information from instructions in [25], whereas sense2vec [26] generated an executable command in robot control language (RCL) [27]. Except for the first study, which addressed the social needs of the elderly, the above research focused on commanding robots to complete tasks through human instructions. These systems were task-oriented and performed well in interactions with brief sentences.

In our case, however, it is necessary to maintain a memorised dialogue to provide users with the robotic scenario information. Therefore, a chatbot is proposed to implement the environment's explainability. Chatbots can be divided into three categories according to their response mechanisms. Information retrieval-based systems [28] are commonly used for frequently asked questions (FAQ) that retrieve standard answers to common questions. Generative-based systems [29], [30] are primarily used in chit-chat chatbots to maintain user conversations. Knowledge graph-based systems [31], [32] are suitable for task-oriented dialogue based on domain knowledge, thus selected in this work.

III. BACKGROUND

The previous section highlighted a KG-based chatbot as a promising method for implementing the system's explainability of real-time robotics scenario information. Generally, there are two approaches to building the knowledge graph: top-down and bottom-up [33]. The top-down approach requires a small group of experts to design top-level and domain-specific ontology patterns as the schema of the KG, such as WordNet [34], and Cyc [35], which is appropriate when there is profound comprehension of the domain's knowledge hierarchy. Conversely, the bottom-up approach derives concepts and relationships from semi-structured data using automated extraction technologies such as DBpedia [36] and YAGO [37], necessitating constant, high-quality data sources for schema management and update.

A. Standardized Ontology for Robotics

We adopt the top-down approach to constructing KGs based on recent research on standardized robotic domain-specific ontologies, such as Core Ontology for Robotics and Automation (CORA)-related ontologies (containing Suggested Upper Merged Ontology (SUMO) [38]-CORA, CORAX, PRARTS and POS) [39], ERAS ontology [40] and Task ontology [41].

- SUMO-CORA: SUMO-CORA ontology is a set of ontologies where the concepts of CORA and SUMO overlap, which defines basic upper-level categories to help us model further progress in robotic scenarios.
- CORA & CORAX & POS & PRARTS: CORA & CORAX ontologies define the majority of concepts in robotics and automation (R&A). CORA ontology includes three general components: *RobotGroup*, *Robot* and *RobotSystem*, whereas CORAX ontology defines some not-so-generic but essential robotics concepts. PRARTS comprises concepts that can represent robot parts. Finally, POS ontology defines concepts for objects' pose, position, and orientation properties.
- ERAS: ERAS ontology considers the ethical usage of robotic techniques based on CORA ontology.
- TO: Task ontology focuses on the task implementation terminology as an extension of CORA ontology.

B. Knowledge Graph Storage

Once the schema of KG is built by ontology, real-world data can be stored graphically. Resource Description Framework (RDF) [42] (e.g., Jena [43]) and labelled property graph (LPG) databases (e.g., Neo4j [44] and ArangoDB*) are emerging technologies for storing graph-structured data [45]. Barrasa identifies three important distinctions between RDF and property graphs [46]: (1) RDF does not uniquely identify instances of the same relationships; (2) RDF does not qualify instances of relationships; (3) RDF can have multivalued properties — triples with the same subject and predicate but for different objects, whereas the LPG only employs arrays for the same purpose. Due to these distinctions, LPG databases provide more efficient storage with compact and intuitive structures optimised for efficient graph traversal.

IV. METHOD

A. System Architecture

To bridge between human operators and robots for the system's explainability in real-time robotics scenario knowledge, we proposed a knowledge graph-based framework including three main modules: conversational agent (CA), knowledge

*ArangoDB homepage

graph (KG) and ROS environment (RE) module, as shown in Figure 1, which aims to help users comprehend the robotic scenario information (e.g., environment and execution information) in natural language.

1) ROS Environment module: In RE module, ROS actions are adopted to generate and execute plans for robotic tasks using actionlib[†] package in Robot Operating System (ROS) environment. The black box is the newly implemented component to obtain raw robotic data, while the blue box was the previous work of the lab [47] for task planning. An Active Inference Server issues tasks to specific action servers (move, look, pick and place) according to the plan made by the Decision Making Agent using active inference [48] after the Perception Server performs the symbolic perception of the current robotic environment. During the execution of the robot's tasks, ROS topics published by action clients and servers will be subscribed by a create_dynamic_kg Node as raw scenario knowledge.

2) Knowledge Graph Module: The robotic scenario data obtained in RE module are stored as graphs in the Neo4j database management system (DBMS) based on the predefined ontology (schema) through Py2neo library [‡]. The ontologies constitute the schema of KG taking advantage of the Neo4j APOC library, which will be described in the next sub-chapter. Notably, users can directly access generated KG graphs in Neo4j Browser, which communicates with the Neo4j DBMS using the Neo4j JavaScript Driver through the Bolt Protocol.

3) Conversational Agent Module: CA model here refers to the chatbot. It is developed using Rasa [49] framework to interpret user queries and extract corresponding information from Neo4j DBMS using Neo4j Python Driver to form appropriate responses. Rasa Open Source is responsible for intent classification and entity recognition supported by NLU Pipeline as well as dialogue management specified by Dialogue Policies. With Rasa SDK, action servers are customized to retrieve relevant information from KGs, and compose events and responses. Finally, the Agent component in Rasa Open Source provides an interface for the above Rasa functionalities and handles a channel to reach users.

B. Knowledge Representation Pipeline

We adopted the knowledge graph as the knowledge representation method since machines can easily store and comprehend real-world data to provide humans with query services due to its efficient and intuitive layout.

1) Intergrated Ontology for Robotics and Automation: Since some standard and well-defined ontologies exist for representing knowledge in a robotic context, we adopted the up-bottom approach to construct knowledge graphs for specifically symbolizing robotic environment knowledge. To form an ontology that takes into account the physical robotic environments, agent plans and robotic tasks, we selected some concepts from CORA-related ontologies (containing SUMO-CORA, CORAX, PRARTS and POS), ERAS ontology and Task ontology to form a Integrated Ontology for Robotics and Automation (IORA), and an overview of the main concepts is shown in Figure 2. Among them, SUMO-CORA defines the general concepts to describe a domain; CORA and CORAX specify the robotic domain; ERAS considers the plan generated by the intelligent agent; and TO further focuses on the task-oriented robotic domain. Using IORA, we could symbolize knowledge in a task-oriented robot scenario. In the next section, the pipeline for adapting CORA-related ontologies in OWL [9] syntax to the graph database will be introduced.

2) Knowledge Storage Pipeline: Compared with RDF data model using knowledge representation languages such as RDFS [50] and OWL, LPG database model is more in line with the logic of people's understanding towards the entities and their relationships in the real world. Besides, it uses arrays to represent multiple properties of subjects and predicts, which equips a more compact structure for efficient user query and understanding. Hence, Neo4j, an object-oriented graph database management system (DBMS) with flexible network structures to store the knowledge of the robotic scenario was adopted, including static environment information and dynamic execution information.

We accessed the OWL file of CORA-related ontologies (SUMO-CORA, CORA, CORAX, PRARTS and POS) through the open source GitHub repository [§] to obtain all the entities and axioms. However, these OWL-based ontologies should be translated to LPG format before adapting them to the schema of knowledge graphs, shown in Figure 2. First, the OWL-based files were imported to Neo4j DMBS by the neosemantics (n10s) - a toolkit that enables the use of RDF and its associated vocabularies in Neo4j. After this operation, the ontologies were stored in the DBMS in the graph format with 147 nodes and 249 relationships.

However, the classes and properties defined in OWL should be translated into nodes and relationships in LPG. The four steps in the following table were carried out to achieve the translation using Neo4j APOC library. Considering the different syntax of OWL and LPG, the main changes were: 1) The default node labels (class description $owl{:}Class$ and object property $owl{:}ObjectProperty$ in OWL)and relationship types (class axioms such as $rdfs{:}subClassOf$ and property axioms $rdfs{:}subPropertyOf$ in OWL) generated after importing ontologies to Neo4j were renamed to be consistent with OWL styles. 2) Property restrictions $owl{:}Restriction$, including value constraints and cardinality constraints on $owl{:}ObjectProperty$ between two $owl{:}Class$ were transformed into two properties $restrictionType$ and $cardinalityVal$ of $owl{:}ObjectProperty$ relationships. 3) To solve the problem that the $rdfs{:}subPropertyOf$ could not be preserved after converting $owl{:}ObjectProperty$ into edges in LPG, we projected $rdfs{:}subPropertyOf$ between $owl{:}ObjectProperty$ into a subgraph separately. 4) Only $owl{:}Class$ will be kept as nodes in LPG while

[†]http://wiki.ros.org/actionlib

[‡]https://py2neo.org/

[§]IEEE1872-owl - GitHub repository

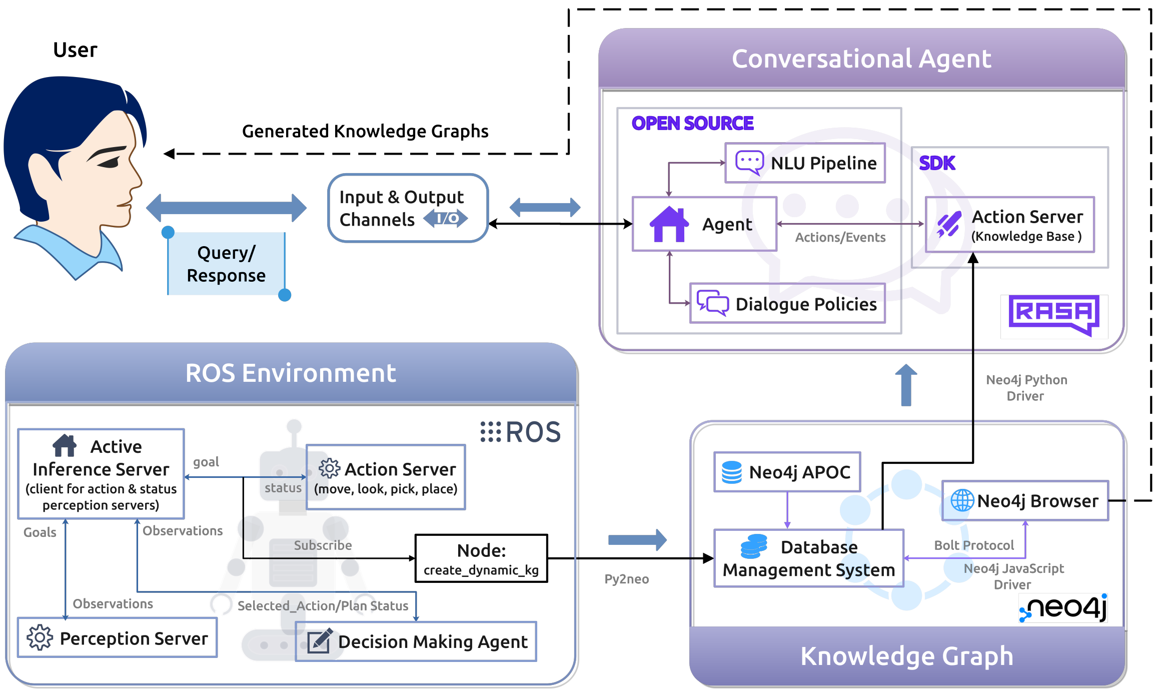


Fig. 1. **System architecture of Iris:** The direction of the arrow in the figure is the direction of information transmission. Black lines illustrate the main workflow between three modules: 1) There are two interactive ways between the user and the system, including communicating with a chatbot and accessing generated knowledge graphs. 2) Robotic data in ROS Environment module is stored in Neo4j DBMS using Py2neo library. 3) Generated knowledge graphs can be accessed by Rasa action servers through Neo4j Python Driver. 4) Users can chat with Iris to query robotic scenario information from KGs either through Input & Output Channels such as Rasa shell and Rasa REST channel.

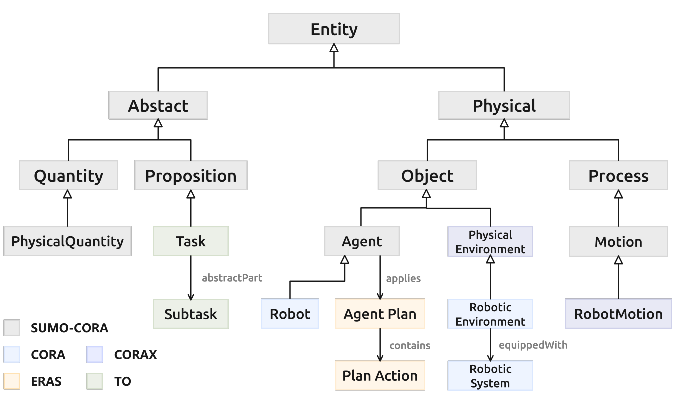


Fig. 2. Overview of the main concepts in our IORA

$owl{:}ObjectProperty$ and $rdfs{:}subClassOf$ will be relationships between nodes. Finally, the translated 92 nodes and 137 relationships were filtered manually along with several necessary concepts from ERAS and TO to form our ICRA-LPG. As it is shown in Figure 4 (only main concepts are included here), ICRA-LPG contains 40 nodes with three properties {name, comment, uri} from {CORA, CORAX, ERAS, SUMO-CORA, TO} labels and 71 relationships with four properties {name, propCharacteristics, restrictionType, uri}, which were ready to model most of the real data for the robotic environment.

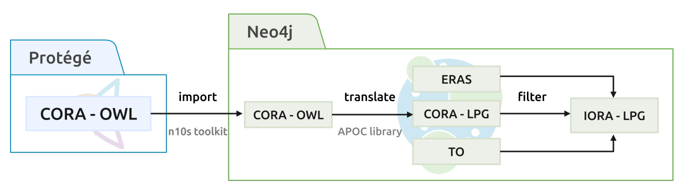


Fig. 3. Pipeline to adapt OWL-based ontologies to LPG-based schema

Translate Rules

Step 1: rename the node labels and relationship types in OWL, such as $n4sch\_\_SCO \rightarrow rdfs{:}subClassOf$, $n4sch\_\_SPO \rightarrow rdfs{:}subPropertyOf$.

Step 2: add property restrictions $owl{:}Restriction$ (value constraint and cardinality constraint) as properties of $owl{:}ObjectProperty$ nodes.

Step 3: project $rdfs{:}subPropertyOf$ relationships between $owl{:}ObjectProperty$ nodes as a subgraph.

Step 4: change the main RDFS construct $(owl{:}Class) \leftarrow [rdfs{:}range]$ - $(owl{:}ObjectProperty)$ - $[rdfs{:}domain] \rightarrow (owl{:}Class)$ to LPG construct $(owl{:}Class)$ - $(owl{:}ObjectProperty) \rightarrow (owl{:}Class)$.

C. Chatbot Pipeline

In our lab, a robotic manipulator with a mobile base is used to perform stacking and picking tasks in a retail setting, shown in Figure 5. We expect that the chatbot could assist

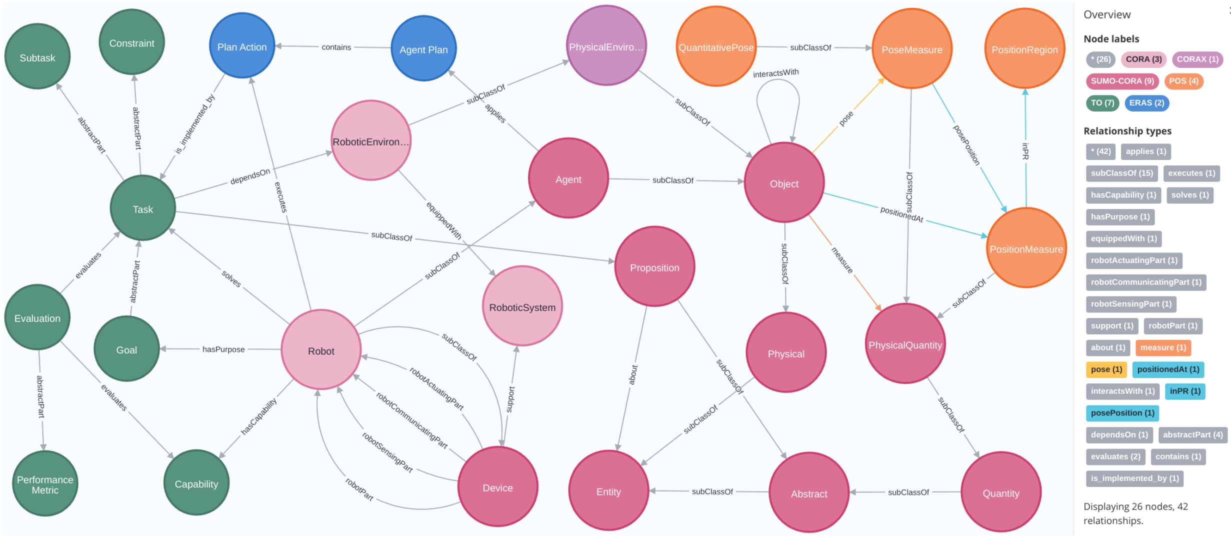


Fig. 4. **ICRA-LPG in Neo4j Browser:** Main 26 concepts and 42 relationships between them from standard ontologies, including CORA-related ontologies(SUMO-CORA, CORA, CORAX and POS), ERAS and TO in different colours are illustrated in the centre of the graph, while legends are displayed on the right.

users in comprehending robotic scenario information, which is anticipated to have the following capabilities:

- Given queries about information of static retailing environment, such as product properties (e.g., mass and position), the chatbot should be able to answer them correctly.
- Given queries about dynamic robot tasks, such as task details and execution status, the chatbot should be able to provide unambiguous responses to users.

Based on these goals, a task-oriented assistant with access to knowledge of the retail environment was designed. Rasa - a machine learning & rule-based framework [49] was adopted as the development platform due to its power NLU pipelines and dialogue policies supported by Rasa Open Source, while flexible, customized action services provided by Rasa SDK. The intents and entities in the user's utterances will be identified by Rasa NLU, following by action services queried the knowledge graphs encoding scenario information to obtain the required information. The assistant will then formulate responses according to dialogue policies [51] and predefined response templates, which reach users through its input & output channels.

1) NLU Pipeline: The following components already embedded in Rasa to ensure the recognition performance in the case of limited training data were selected for this study: (1) the pre-trained language model SpacyNLP [52] including SpacyTokenizer and SpacyFeaturizer are used for tokenization and word embedding; (2) DIETClassifier [53] is adopted for intent classification and entity extraction. Besides, (3) the default FallbackClassifier allows the assistant to handle incoming messages with low NLU confidence, while (4) EntitySynonymMapper helps the system map recognized entities to predefined entities. the primary intents and some training examples fed into Rasa was shown in Table I. The intentions

Fig. 5. The retail environment in our Lab

above the horizontal line are for the users' inquiries about some pre-stored and static environmental knowledge, such as the products and their properties in stores, while others are to help people understand the changing and dynamic execution information when the robot is doing specific tasks, including queries about products' location and task/action status. The content in square brackets is a training example, and the content in parentheses specifies its entity type. To store entity values recognized from users' utterances, We defined 7 entity types, including $object\_type$, $object$, $attribute$, $property$, $furniture$, $task$ and $subtask$.

2) Action Server: Several action servers were customized to query Neo4j DBMS in [54] language through Py2neo library and dispatch appropriate messages based on predefined answering templates to the user. The functional action servers and corresponding query templates in Cypher are shown in Table II. The actions above the horizontal line query the knowledge graph that stores knowledge about products, whereas the actions below query the knowledge graph that

TABLE I
NLU TRAINING DATA EXAMPLES

Intent	Training example
query_environment	Give me some background on AIRLab.
introduce_chatbot	Who am I talking with?
chatbot_capability	What can you do?
query_product_in_env	What [items](object_type) are in the environment?
query_product_property	Tell me [mass](attribute) of [yogurt](object)
	Do the environment contains [flower](object)?
query_specific_product	Tell me [more](property) about [yogurt](object).
	What type of [properties](property) can I ask?
query_product_location	where I could find [juice](object)?
query_product_furniture	What products on the [shelf_1](furniture)?
query_specific_task	I want to know about [task2](task).
query_current_task	What is the current task?
query_previous_task	Give me an overview of previous tasks.
query_specific_action	tell me [first subtask](subtask) of [task_3](task).
query_current_action	What are you doing now?
query_previous_action	What did you do in the last subtask?

stores information about tasks.

TABLE II
NLU TRAINING DATA EXAMPLES

Action server	Cypher template
product_in_env	MATCH (o:{object_type} {attrs}) RETURN o
product_property specific_product	MATCH (o:Object) WHERE o.name=? RETURN properties(o)
product_location	MATCH (o:Object)-[r*2]- (p:PositionRegion) WHERE o.name=?
product_furniture	MATCH (o:Object)-[r*2]- (p:PositionRegion) WHERE p.value=?
specific_task	MATCH (t:Task)-[r*2]-(a:'Agent Plan') WHERE t.name=? RETURN t,a
current_task	MATCH (t:Task)-[r*2]-(a:'Agent Plan') Where t.status='ACTIVE'
previous_task	MATCH (t:Task)-[r*2]-(a:'Agent Plan') Where p.status='SUCCEEDED'
specific_action	MATCH (a:'Agent Plan')-[r*3]- (s:Subtask) WHERE s.name and s.partof
current_action	MATCH (a:'Agent Plan')-[r*3]- (s:Subtask) WHERE s.status='ACTIVE'
previous_action	MATCH (s1:Subtask)-[r*2]-(s2:Subtask) WHERE s1.status='ACTIVE'

V. Experiments

To evaluate the effectiveness of Iris, participants were invited to conduct the designed experiments, including two simulated scenarios: Scenario 1 and Scenario 2.

A. Design

The first scenario was static, containing information on 6 products in stimulated retail environment, as shown in Figure 6. The corresponding concepts and relationships which related to product information from ICRA-LPG are selected as the schema of real-world data. The light green nodes are products with $Object$ label in the environment, the blue nodes store their position properties with labels $PositionMeasure$ (position in global coordinate) and $PositionRegion$ (region in the environment); while dark green nodes are corresponding poses to grab those products with $PoseMeasure$. The right bottom of the graph provides an example of detailed properties of milk, such as its mass is 1.2 kg, and its position is located in Table2. While the second scenario was dynamic, containing three robotics tasks: 1) the robot is at the place where milk is located; 2) the robot holds hagelslag and 3) the tea_box is in the robot's basket. To finish the above tasks, the robot will continuously perform some subtasks (actions). As it is shown in Figure 7, concepts $Agent$, $AgentPlan$, $PlanAction$, $Task$ and $Subtask$ from ICRA-LPG store information of plan generated by the decision-making agent and task execution. The right bottom of this graph also gives an example of plan execution information. To reach the final state *the tea_box is in the basket* of task_3, the plan contains three actions *move to reach tea_box*, *pick tea_box* and *place it in the basket*. Therefore, $subtask\_3$ is the last action $place\_in\_basket$ of task_3.

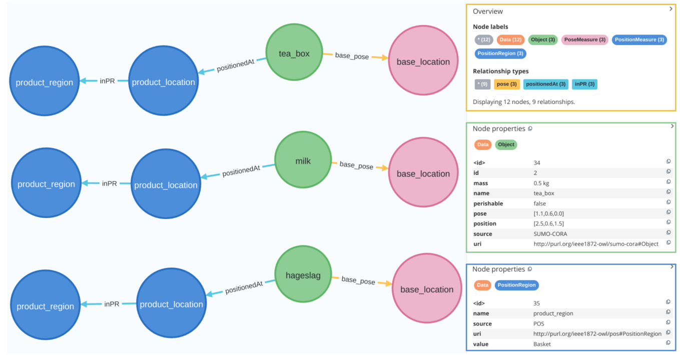


Fig. 6. Scenario 1: static product information in KG

We recruited four groups of participants with a robotics background (approximately 5 to 15 people per group), Group 1: only look at the real-time output information about detailed task execution in Linux terminals when running three robotic tasks mentioned above (duration of this procedure is 5 minutes); Group 2: only interact with Iris to obtain static retailing environment and dynamic robot task information (duration of each procedure is 5 minutes, so 10 mins in total); Group 3: only look at pre-stored static KG and dynamically generated KG to access the same scenario information as Group 2; Group 4: interact with chatbot (the same as Group 2) & look at the two KGs to understand scenario information (duration of this procedure is 5 minutes). The participants' comprehension of the two robotic scenarios varied based on the intuitiveness of each interaction procedure they encountered. Therefore, some hypotheses were made as follows:

- **Hypothesis 1**: The information that users only get from the terminal is limited, so they cannot understand the three tasks very clearly, while interacting with Iris or looking at knowledge graphs will help users comprehend the two scenarios.
- **Hypothesis 2**: Users who are exposed to Iris and knowledge graphs will have the deepest understanding towards the two scenarios.

The questionnaire shown in Table VI, Table VII and Table VIII was designed to verify hypotheses based on their memory of what they obtained during the interaction. The questionnaire is divided into two parts: 1) about the static retailing environment and 2) about the dynamic robot tasks. In terms of the static retailing environment, people were be asked to remember and fill the mass, ID, pose location, position and

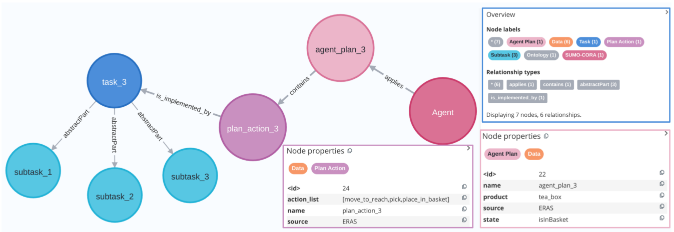
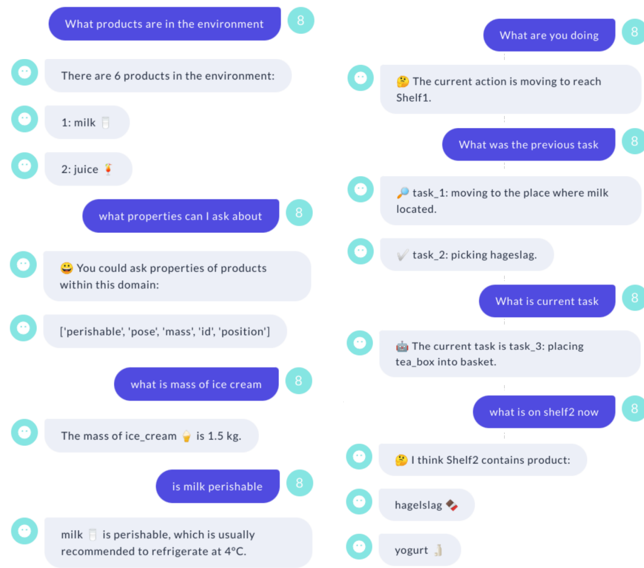


Fig. 7. Scenario 2: dynamic execution information in KG



(a) Chat history of Scenario 1 (b) Chat history of Scenario 2

Fig. 8. Chat history of static information & dynamic information

perishable of six products in the simulated scenarios, while the dynamic tasks' part includes the subtasks of each task, the order of tasks, objects of tasks and the final location of products after all the tasks. The Linux group only filled the second part and is compared with the other group's second part results. Furthermore, an additional questionnaire is required for those groups chatting with the Rasa assistant to evaluate the chatbot's performance.

B. Results

In total, 33 participants were invited for the experiments, 7 people for Linux, 11 for chatbot, 7 for KG, and 8 for KG plus chatbot. Some examples of chat history in Scenario 1 and Scenario 2 from Group 2 and Group 4 are shown in Figure 8.

To evaluate the repeatability and the consistency of different groups for the same quantitative experiment, the Intraclass correlation coefficient (ICC) [55] was introduced here. The result is given in Table. III. The four groups, except for the KG, are larger than 0.75 with confidence $p < 0.001$, which proves the good repeatability in the three groups. The KG group is marginally lower with ICC 0.72 and confidence $p < 0.001$, achieving decent repeatability. The differences come from that the people in KG group are asked to remember two knowledge graphs in total 10 minutes, thus some may focus more on static question while some not.

TABLE III
THE EVALUATION OF THE CONSISTENCY FOR FOUR GROUPS

Group Index (num of people)	ICC	Confidence probability
Linux (7)	0.8998	$p < 4.61e-7$
Chatbot (11)	0.7931	$p < 5.41e-6$
KG (7)	0.7242	$p < 3.07e-4$
KG plus chatbot (8)	0.8701	$p < 6.35e-10$

The accuracy of participants' answers to questions in the questionnaire in Scenario 1 and Scenario 2 are calculated to evaluate the effectiveness of the proposed Iris. It is calculated with a weighted sum to compensate for the different difficulties of the questions, i.e., the position with a higher weight than the perishable. The results are given in Table. IV. The Linux group only has access to dynamic tasks, so it is compared with other groups within this field, and achieving the worst result as in hypothesis 1. The chatbot outperforms the Linux group slightly in the dynamic experiments because some participants are chatting with the chatbot instead of focusing on the goal, as mentioned. The KG group has higher accuracy than the chatbot group in the dynamic test, while it is reversed in the static test. This is partly expected as the question asked for the dynamic test is directly related to the knowledge graph, while the chatbot group needs extra effort to combine the asked question to get the answer. Another reason is that some questions are not even asked by the participant but are still calculated here for a fair comparison to maintaining the total weight sum equal to one. The proposed KG plus chatbot group achieves better results compared with the chatbot and the KG groups, as we expected, which proves our proposed pipeline for assisting humans in AIRlab retail setting.

TABLE IV
THE EVALUATION OF THE ACCURACY FOR FOUR GROUPS

Type	Group	Accuracy
Static	Chatbot	0.2915
	KG	0.2868
	KG plus chatbot	0.4407
Dynamic	Linux	0.3915
	chatbot	0.4512
	KG	0.6323
	KG plus chatbot	0.6821

To evaluate the performance of the proposed chatbot, the PARAdigm for DIalogue System Evaluation (PARADISE) [56], as one of the most extensively utilized frameworks for combining different levels of evaluation is used here. The weights for the individual questionnaire questions for different subjective variables are set as (i) system usability 0.25, (ii) clarity 0.25, (iii) naturalness 0.05, (iv) friendliness 0.1, (v) robustness to misunderstandings 0.2, and (vi) willingness to use the system again 0.15, corresponding to the designed questionnaire directly. The answers for each question from strongly disagree and strongly agree are projected to the interval $[-3, 3]$.

$$P = (\alpha * \mathcal{N}(k)) - \sum_i w_i * \mathcal{N}(c_i) \tag{1}$$

where P is the performance score calculated with the weighted answer from the questionnaire, k is obtained confusion matrix, designed with intent and entity of the chatbot performances, with each weighted 0.55 and 0.45. $c_i, i \in [1:4]$ is the cost, includes the total number of system and users' turns, time per turn, reprompt numbers, inappropriate response, α and w_i is the regressed weight, and $\mathcal{N}$ is a Z score normalization.

The evaluations are based on different types of tasks and different groups, as shown in Table. V. The KG plus chatbot achieves better performance than only chatbot because it will be helpful if people intuitively understand the working principle behind the chatbot. Also, the dynamic task is more complicated than the static one, and it is expected that the score will decrease in the chatbot group. On the other hand, it is interesting to see that the performances for the KG plus chatbot group are highly improved, meaning that understanding KG will help people to tackle complex tasks.

TABLE V
THE EVALUATION OF THE ACCURACY FOR FOUR GROUPS

Type	Group	People number	Performances
Static	Chatbot	4	2.80e-16
	KG plus chatbot	7	1.59e-15
Dynamic	Chatbot	5	1.97e-16
	KG plus chatbot	7	1.84e-15

VI. CONCLUSION

In this paper, we developed a KG-based chatbot to assist users in comprehending robotic scenario information, including static environmental information and dynamic task information. The proposed chatbot is able to explain the complex environmental knowledge to users in natural language or by KG compared with other NLP technology only commands the robot to perform specific tasks in HRI. Based on the background of AIRLab Delft, several IEEE standard ontologies are extracted and ICRA-LPG schema is adapted as a task-oriented ontology for a retail setting.

A case study was conducted, and a significant improvement can be obtained with our designed system, at least 1.5 times improvement for both static and dynamic scenarios. It is noted that the proposed task-oriented ontology plays a great role in assisting users with dynamic tasks. The proposed chatbot could be enhanced to update static information and command the robot to complete tasks in a bidirectional approach in future work. In addition, the potential enhancement of human trust towards the robotic environment provided by this system can also be a meaningful research direction.

ACKNOWLEDGMENT

REFERENCES

[1] U. Ehsan, Q. V. Liao, M. Muller, M. O. Riedl, and J. D. Weisz, "Expanding explainability: Towards social transparency in ai systems," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021, pp. 1–19.

[2] W. Pieters, "Explanation and trust: what to tell the user in security and ai?" *Ethics and information technology*, vol. 13, no. 1, pp. 53–64, 2011.

[3] S. Anjomshoae, A. Najjar, D. Calvaresi, and K. Främling, "Explainable agents and robots: Results from a systematic literature review," in *18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), Montreal, Canada, May 13–17, 2019*. International Foundation for Autonomous Agents and Multiagent Systems, 2019, pp. 1078–1088.

[4] B. Hayes and J. A. Shah, "Improving robot controller transparency through autonomous policy explanation," in *Proceedings of the 2017 ACM/IEEE international conference on human-robot interaction*, 2017, pp. 303–312.

[5] M. Edmonds, F. Gao, H. Liu, X. Xie, S. Qi, B. Rothrock, Y. Zhu, Y. N. Wu, H. Lu, and S.-C. Zhu, "A tale of two explanations: Enhancing human trust by explaining robot behavior," *Science Robotics*, vol. 4, no. 37, p. eaay4663, 2019.

[6] Z. Han, D. Giger, J. Allspaw, M. S. Lee, H. Admoni, and H. A. Yanco, "Building the foundation of robot explanation generation using behavior trees," *ACM Transactions on Human-Robot Interaction (THRI)*, vol. 10, no. 3, pp. 1–31, 2021.

[7] Z. Han, E. Phillips, and H. A. Yanco, "The need for verbal robot explanations and how people would like a robot to explain itself," *ACM Transactions on Human-Robot Interaction (THRI)*, vol. 10, no. 4, pp. 1–42, 2021.

[8] J. L. Wright, J. Y. Chen, and S. G. Lakhmani, "Agent transparency and reliability in human–robot interaction: The influence on user confidence and perceived reliability," *IEEE Transactions on Human-Machine Systems*, vol. 50, no. 3, pp. 254–263, 2019.

[9] D. L. McGuinness, F. Van Harmelen *et al.*, "Owl web ontology language overview," *W3C recommendation*, vol. 10, no. 10, p. 2004, 2004.

[10] M. A. Rodriguez and P. Neubauer, "Constructions from dots and lines," *arXiv preprint arXiv:1006.2361*, 2010.

[11] ——, "The graph traversal pattern," in *Graph data management: Techniques and applications*. IGI global, 2012, pp. 29–46.

[12] T. B. Sheridan, "Human–robot interaction: status and challenges," *Human factors*, vol. 58, no. 4, pp. 525–532, 2016.

[13] M. A. Salichs, R. Barber, A. M. Khamis, M. Malfaz, J. F. Gorostiza, R. Pacheco, R. Rivas, A. Corrales, E. Delgado, and D. Garcia, "Maggie: A robotic platform for human-robot social interaction," in *2006 IEEE conference on robotics, automation and mechatronics*. IEEE, 2006, pp. 1–7.

[14] M. Marge, C. Espy-Wilson, and N. Ward, "Spoken language interaction with robots: Research issues and recommendations, report from the nsf future directions workshop," *arXiv preprint arXiv:2011.05533*, 2020.

[15] M. Faridghasemnia, D. Nardi, and A. Saffiotti, "Towards abstract relational learning in human robot interaction," *arXiv preprint arXiv:2011.10364*, 2020.

[16] K. Jokinen, S. Nishimura, K. Watanabe, and T. Nishimura, "Human-robot dialogues for explaining activities," in *9th International Workshop on Spoken Dialogue System Technology*. Springer, 2019, pp. 239–251.

[17] L. Grassi, C. T. Recchiuto, and A. Sgorbissa, "Knowledge triggering, extraction and storage via human–robot verbal interaction," *Robotics and Autonomous Systems*, vol. 148, p. 103938, 2022.

[18] B. Bruno, C. T. Recchiuto, I. Papadopoulos, A. Saffiotti, C. Koulouglioti, R. Menicatti, F. Mastrogiovanni, R. Zaccaria, and A. Sgorbissa, "Knowledge representation for culturally competent personal robots: requirements, design principles, implementation, and assessment," *International Journal of Social Robotics*, vol. 11, pp. 515–538, 2019.

[19] A. Singhal, "Introducing the knowledge graph: things, not strings," [EB/OL], 2012, https://blog.google/products/search/introducing-knowledge-graph-things-not/.

[20] R. Reinanda, E. Meij, M. de Rijke *et al.*, *Knowledge graphs: An information retrieval perspective*. Now Publishers, 2020.

[21] B. Schrader, "What is the difference between an ontology and a knowledge graph," 2020.

[22] L. He, K. Lee, M. Lewis, and L. Zettlemoyer, "Deep semantic role labeling: What works and what's next," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, pp. 473–483.
[23] M. Sukhwani, V. Duggal, and S. Zahrai, "Dynamic knowledge graphs as semantic memory model for industrial robots," *arXiv preprint arXiv:2101.01099*, 2021.
[24] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," 2001.
[25] Z. Li, Y. Mu, Z. Sun, S. Song, J. Su, and J. Zhang, "Intention understanding in human–robot interaction based on visual-nlp semantics," *Frontiers in Neurorobotics*, vol. 14, p. 610139, 2021.
[26] A. Trask, P. Michalak, and J. Liu, "sense2vec-a fast and accurate method for word sense disambiguation in neural word embeddings," *arXiv preprint arXiv:1511.06388*, 2015.
[27] C. Matuszek, E. Herbst, L. Zettlemoyer, and D. Fox, "Learning to parse natural language commands to a robot control system," in *Experimental robotics: the 13th international symposium on experimental robotics*. Springer, 2013, pp. 403–415.
[28] Z. Ji, Z. Lu, and H. Li, "An information retrieval approach to short text conversation," *arXiv preprint arXiv:1408.6988*, 2014.
[29] A. Ritter, C. Cherry, and B. Dolan, "Data-driven response generation in social media," in *Empirical Methods in Natural Language Processing (EMNLP)*, 2011.
[30] L. Shang, Z. Lu, and H. Li, "Neural responding machine for short-text conversation," *arXiv preprint arXiv:1503.02364*, 2015.
[31] D. Kadariya, R. Venkataramanan, H. Y. Yip, M. Kalra, K. Thirunarayanan, and A. Sheth, "kbot: knowledge-enabled personalized chatbot for asthma self-management," in *2019 IEEE International Conference on Smart Computing (SMARTCOMP)*. IEEE, 2019, pp. 138–143.
[32] A. Ait-Mlouk and L. Jiang, "Kbot: a knowledge graph based chatbot for natural language understanding over linked data," *IEEE Access*, vol. 8, pp. 149 220–149 230, 2020.
[33] X. Hao, Z. Ji, X. Li, L. Yin, L. Liu, M. Sun, Q. Liu, and R. Yang, "Construction and application of a knowledge graph," *Remote Sensing*, vol. 13, no. 13, p. 2511, 2021.
[34] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.
[35] D. B. Lenat, "Cyc: A large-scale investment in knowledge infrastructure," *Communications of the ACM*, vol. 38, no. 11, pp. 33–38, 1995.
[36] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "Dbpedia: A nucleus for a web of open data," in *The semantic web*. Springer, 2007, pp. 722–735.
[37] F. M. Suchanek, G. Kasneci, and G. Weikum, "Yago: A large ontology from wikipedia and wordnet," *Journal of Web Semantics*, vol. 6, no. 3, pp. 203–217, 2008.
[38] A. Pease, I. Niles, and J. Li, "The suggested upper merged ontology: A large ontology for the semantic web and its applications," in *Working notes of the AAAI-2002 workshop on ontologies and the semantic web*, vol. 28, 2002, pp. 7–10.
[39] "Ieee standard ontologies for robotics and automation," *IEEE Std 1872-2015*, pp. 1–60, 2015.
[40] I. S. Association *et al.*, "P7007—ontological standard for ethically driven robotics and automation systems," 2017.
[41] S. Balakirsky, C. Schlenoff, S. Rama Fiorini, S. Redfield, M. Barreto, H. Nakawala, J. L. Carbonera, L. Soldatova, J. Bermejo-Alonso, F. Maikore *et al.*, "Towards a robot task ontology standard," in *International Manufacturing Science and Engineering Conference*, vol. 50749. American Society of Mechanical Engineers, 2017, p. V003T04A049.
[42] O. Lassila, R. R. Swick *et al.*, "Resource description framework (rdf) model and syntax specification," 1998.
[43] B. McBride, "Jena: A semantic web toolkit," *IEEE Internet computing*, vol. 6, no. 6, pp. 55–59, 2002.
[44] I. Robinson, J. Webber, and E. Eifrem, *Graph databases: new opportunities for connected data*. " O'Reilly Media, Inc.", 2015.
[45] D. Alocci, J. Mariethoz, O. Horlacher, J. T. Bolleman, M. P. Campbell, and F. Lisacek, "Property graph vs rdf triple store: A comparison on glycan substructure search," *PloS one*, vol. 10, no. 12, p. e0144578, 2015.
[46] J. Barrasa, "Rdf triple stores vs. labeled property graphs: What's the difference?" [EB/OL], 2017, https://neo4j.com/blog/rdf-triple-store-vs-labeled-property-graph-difference/.
[47] C. Pezzato, C. H. Corbato, S. Bonhof, and M. Wisse, "Active inference and behavior trees for reactive action planning and execution in robotics," *IEEE Transactions on Robotics*, 2023.
[48] K. Friston, T. FitzGerald, F. Rigoli, P. Schwartenbeck, G. Pezzulo *et al.*, "Active inference and learning," *Neuroscience & Biobehavioral Reviews*, vol. 68, pp. 862–879, 2016.
[49] T. Bocklisch, J. Faulkner, N. Pawlowski, and A. Nichol, "Rasa: Open source language understanding and dialogue management," *arXiv preprint arXiv:1712.05181*, 2017.
[50] D. Brickley, R. V. Guha, and A. Layman, "Resource description framework (rdf) schema specification," Technical report, W3C, 1999. W3C Proposed Recommendation. http://www. w3 . . . , Tech. Rep., 1998.
[51] V. Vlasov, J. E. Mosig, and A. Nichol, "Dialogue transformers," *arXiv preprint arXiv:1910.00486*, 2019.
[52] M. Honnibal and I. Montani, "spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing," *To appear*, vol. 7, no. 1, pp. 411–420, 2017.
[53] T. Bunk, D. Varshneya, V. Vlasov, and A. Nichol, "Diet: Lightweight language understanding for dialogue systems," *arXiv preprint arXiv:2004.09936*, 2020.
[54] N. Francis, A. Green, P. Guagliardo, L. Libkin, T. Lindaaker, V. Marsault, S. Plantikow, M. Rydberg, P. Selmer, and A. Taylor, "Cypher: An evolving query language for property graphs," in *Proceedings of the 2018 International Conference on Management of Data*, 2018, pp. 1433–1445.
[55] J. J. Bartko, "The intraclass correlation coefficient as a measure of reliability," *Psychological reports*, vol. 19, no. 1, pp. 3–11, 1966.
[56] M. A. Walker, D. J. Litman, C. A. Kamm, and A. Abella, "Paradise: A framework for evaluating spoken dialogue agents," 1997. [Online]. Available: https://arxiv.org/abs/cmp-lg/9704004

APPENDIX

TABLE VI
QUESTIONNAIRE FOR CHATBOT EVALUATION: GROUP2 & GROUP4

Question List		Strongly disagree						Strongly agree
Item		1	2	3	4	5	6	7
1	The chatbot responds too slowly.	◯	◯	◯	◯	◯	◯	◯
2	It was easy to lose track of where you are in the interaction.	◯	◯	◯	◯	◯	◯	◯
3	It is easy to learn how to use the chatbot.	◯	◯	◯	◯	◯	◯	◯
4	The chabot's responses were accurate.	◯	◯	◯	◯	◯	◯	◯
5	The chatbot didn't always do what I wanted.	◯	◯	◯	◯	◯	◯	◯
6	The chatbot was organized and logical.	◯	◯	◯	◯	◯	◯	◯
7	The chatbot was understandable.	◯	◯	◯	◯	◯	◯	◯
8	The interaction with the chatbot was consistent.	◯	◯	◯	◯	◯	◯	◯
9	The chatbot used everyday words.	◯	◯	◯	◯	◯	◯	◯
10	The chatbot's response sounded enthusiastic.	◯	◯	◯	◯	◯	◯	◯
11	I felt comfortable using the system.	◯	◯	◯	◯	◯	◯	◯
12	The chatbot seemed friendly.	◯	◯	◯	◯	◯	◯	◯
13	I was able to recover easily from errors.	◯	◯	◯	◯	◯	◯	◯
14	The chatbot made a few errors.	◯	◯	◯	◯	◯	◯	◯
15	I felt in control of the interaction with the chatbot.	◯	◯	◯	◯	◯	◯	◯
16	I would be likely to use this system again.	◯	◯	◯	◯	◯	◯	◯
17	The system was useful.	◯	◯	◯	◯	◯	◯	◯
18	The chatbot would help me be more productive.	◯	◯	◯	◯	◯	◯	◯

TABLE VII
QUESTIONNAIRE FOR STATIC PRODUCT INFORMATION: GROUP 2 & GROUP 3 & GROUP4

Item	ID	mass	Pose to grab	Position	Perishable
Milk					
Tea box					
Hageslag					
Yogurt					
Juice					
Ice cream					

TABLE VIII
QUESTIONNAIRE FOR DYNAMIC ROBOT TASKS: GROUP 1 & GROUP 2 & GROUP 3 & GROUP4

1	What is robot's first task ?	◯ Move		◯ Pick		◯ Place		◯ Not sure
2	What is robot's second task ?	◯ Move		◯ Pick		◯ Place		◯ Not sure
3	What is robot's third task ?	◯ Move		◯ Pick		◯ Place		◯ Not sure
4	Which products are robot's first task ?	◯ Milk	◯ Tea box	◯Hageslag	◯ Yogurt	◯ Juice	◯ Ice cream	◯ Not sure
5	Which products are robot's second task ?	◯ Milk	◯ Tea box	◯Hageslag	◯ Yogurt	◯ Juice	◯ Ice cream	◯ Not sure
6	Which products are robot's third task ?	◯ Milk	◯ Tea box	◯Hageslag	◯ Yogurt	◯ Juice	◯ Ice cream	◯ Not sure
7	Which products are on the Table 1 at the end?	◯ Milk	◯ Tea box	◯Hageslag	◯ Yogurt	◯ Juice	◯ Ice cream	◯ Not sure
8	Which products are on the Table 2 at the end?	◯ Milk	◯ Tea box	◯Hageslag	◯ Yogurt	◯ Juice	◯ Ice cream	◯ Not sure
9	Which products are on the Shelf 1 at the end?	◯ Milk	◯ Tea box	◯Hageslag	◯ Yogurt	◯ Juice	◯ Ice cream	◯ Not sure
9	Which products are on the Shelf 2 at the end?	◯ Milk	◯ Tea box	◯Hageslag	◯ Yogurt	◯ Juice	◯ Ice cream	◯ Not sure
10	What's your age	Please write your answer:						
	level	1	2	3	4	5	6	7
10	What your professional level with ROS	◯	◯	◯	◯	◯	◯	◯
11	What your professional level with Knowledge graph	◯	◯	◯	◯	◯	◯	◯