

# Blending ontologies and language models to generate sound and natural robot explanations

Alberto Olivares-Alarcos  
Institut de Robòtica i Informàtica  
Industrial, CSIC-UPC  
Barcelona, Spain  
aolivares@iri.upc.edu

Muhammad Ahsan  
Institute of Electrical and Control  
Engineering, National Yang Ming  
Chiao Tung University  
Hsinchu, Taiwan  
muhammadahsanfs.ee12@nycu.edu.tw

Satrio Sanjaya  
Institute of Electrical and Control  
Engineering, National Yang Ming  
Chiao Tung University  
Hsinchu, Taiwan  
satriosanjaya.ee11@nycu.edu.tw

Hsien-I Lin  
Institute of Electrical and Control  
Engineering, National Yang Ming  
Chiao Tung University  
Hsinchu, Taiwan  
sofin@nycu.edu.tw

Guillem Alenyà  
Institut de Robòtica i Informàtica  
Industrial, CSIC-UPC  
Barcelona, Spain  
aolivares@iri.upc.edu

## ABSTRACT

Building effective human-robot interaction requires robots to derive conclusions from their experiences that are both logically sound and communicated in ways aligned with human expectations. This paper presents a hybrid framework that blends ontology-based reasoning with large language models (LLMs) to produce semantically grounded and natural robot explanations. Ontologies ensure logical consistency and domain grounding, while LLMs provide fluent, context-aware and adaptive language generation. The proposed method grounds data from human-robot experiences, enabling robots to reason about whether events are typical or atypical based on their properties. We integrate a state-of-the-art algorithm for retrieving and constructing contrastive ontology-based narratives with an LLM agent that refines them into concise, clear explanations. The approach is validated through a laboratory study replicating an industrial collaborative task. Empirical results show significant improvements in the clarity and brevity of ontology-based narratives while preserving their semantic accuracy. Overall, this work highlights the potential of ontology-LLM integration to advance explainable agency, enhance transparency, and promote more intuitive human-robot collaboration.

## KEYWORDS

applied ontology, explainable robots, foundation models, collaborative robotics, contrastive explanations

### ACM Reference Format:

Alberto Olivares-Alarcos, Muhammad Ahsan, Satrio Sanjaya, Hsien-I Lin, and Guillem Alenyà. 2026. Blending ontologies and language models to generate sound and natural robot explanations. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026)*, Paphos, Cyprus, May 25 – 29, 2026, IFAAMAS, 10 pages.

---

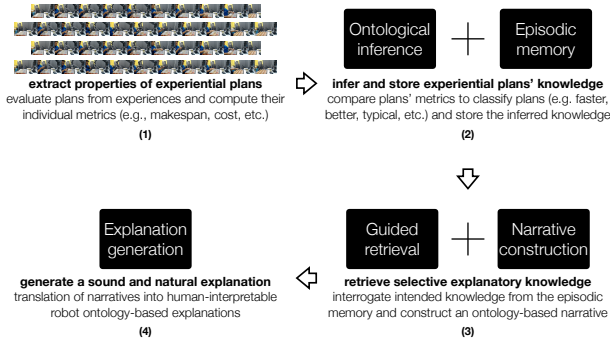
*Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026)*, C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus. © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). This work is licensed under the Creative Commons Attribution 4.0 International (CC-BY 4.0) licence.

## 1 INTRODUCTION

Bidirectional communication between interactive agents is a fundamental requirement for successful social coordination, particularly in collaborative human-robot scenarios [43]. The emergence of large language models (LLMs) and other foundation models offers new opportunities to enable interactive information exchange between humans and robots [44]. For example, those models may contribute to the field of explainable agency (explaining the reasoning behind the behavior of goal-driven agents and robots) [2]. However, foundation models often lack awareness of domain-specific knowledge and internal robot states, and their tendency to generate fluent yet ungrounded text can lead to hallucinations, misleading outputs, and false information. This is especially crucial in long-horizon tasks, where robots are expected to work for extended periods of time and the quantity of internal knowledge is huge. Retrieval-augmented generation (RAG) emerges as a promising solution to this drawback [13, 16], using domain-specific knowledge sources to improve the reliability of explanations by aligning them with the contextual knowledge of the robot.

Knowledge representation formalisms, such as ontologies, are great sources of domain-specific knowledge. Ontologies provide a formal and semantically rich representation of knowledge, making them a natural fit for the retrieval process within RAG-based explainable systems. Indeed, some authors argue that explainable systems must rest on explicit and reliable representations, which are only attainable through robust semantic and ontological frameworks. In essence, explanation presupposes semantics, and semantics presupposes ontology [18]. Such an argument is empirically supported by a solid literature on ontology-based explainable artificial intelligence systems [28, 33, 38].

In the robotics domain, the development and use of ontologies have experienced stable growth for more than a decade [4, 5, 14, 17, 27, 30, 34, 37, 42]. Recent works have even investigated the generation of ontology-based robot explanatory narratives, claiming positive insights and also pointing out some limitations (e.g. ontology-based narratives tend to be verbose and complex to read) [29, 31]. Hence, the time is ripe to embrace ontology-informed language



**Figure 1: Overview of the methodology for generating sound and natural explanations blending ontologies and LLMs. (1) and (2) operate on  $n$  plans, while (3) and (4) focus on 2 plans (e.g., current and previous executions, current and prototypical, ...).**

models as a foundation for explainable robots. The combination of LLMs, RAG and ontologies may create a robust framework for explainable robots, where LLMs generate content, RAG maintains reliability, and ontologies provide the semantic backbone, ensuring that explanations are grounded and semantically coherent.

Building on these considerations, this work investigates the following research question: *How can robots model and reason about recurring patterns in their experiences and generate explanations that compare them in a sound and natural manner?* This article addresses the question by presenting a novel methodology (see Fig. 1) for generating robot ontology-based explanations. The methodology integrates a method to analyze data from robot experiences and extract common execution plan properties, a framework to generate robot memories of the extracted information, a state-of-the-art approach to retrieving *sound* ontology-based robot experience narratives [32], and an LLM agent for generating *natural* robot explanations of how the experiences compare. Our approach is validated with robot experiences collected using a laboratory mock-up task of a real industrial scenario in which a human and a robot collaborate to inspect the surface of solid-state drive (SSD) cases. The generated explanations, which compare the different experiences, are evaluated against the state-of-the-art method for ontology-based narratives (baseline). The performance of both methods is statistically compared using a set of objective evaluation metrics: explanation length (brevity), readability score (clarity), and semantic similarity (semantic coherence). The proposed method outperforms the baseline, producing shorter and clearer explanations while preserving the semantic meaning and coherence of the original narratives. It is also discussed how the novel methodology can adapt to user interactions asking for changes in the generated explanation: a shorter or clearer one.

## 2 RELATED WORK

**Sound formalisms for explainable robots:** Flores et al. [15], proposed a method to transform the task history record of a robot, which is written in a declarative logical language for robot programming (SitLog), into a narrative knowledge representation. The

literature also comprises some efforts to verbalize or narrate robot tasks knowledge encoded using symbolic planning formalisms (e.g. PDDL, RDDL) [9, 36]. The narratives described a plan’s sequence, the rationale to include a task in the plan, etc. Sridharan’s work [41], combined non-monotonic logic, capable of supporting abductive reasoning, with deep learning, exploiting their complementary strengths in representation, reasoning, and learning to implement explainable agency. Some works leveraged the structure of ontology-based robot episodic memories (the collection of past personal experiences that occurred at particular times and places [12]) to retrieve past experiences knowledge to construct sound robot explanatory narratives [29, 32]. Some researchers exploited behavior trees, a mathematical model of plan execution, to produce robot explanations. Han et al. [19], framed the specific format of behavior trees into a new structure comprising the goal of the behavior tree, a set of subgoals, abstract steps to achieve the subgoals, and the final atomic actions. Depending on the target question to answer, a set of template-based explanations are generated. Love et al. [24], presented a method capable of generating causal and counterfactual explanations in response to contrastive ‘why’ questions to clarify the rationale behind robot decisions and actions. Those works rely on the natural robustness and interpretability of their respective formalism to produce sound narratives or explanations. However, while the generated narratives can be easily interpreted by experts (e.g. actual robot designers), it may not be so easy for final users. In most cases, the narratives contain truthful information, but in a verbose and complex format that hinders readability and understandability. Furthermore, once the narrative is generated, there is no possible interaction between the explainee (explanation’s receiver) and the explainer (e.g. asking for more details).

**LLMs for explainable robots:** The literature is currently rich in works on generating robot explanations using LLMs. Often, those works use RAG architectures to manipulate information that is relevant for the execution of robot tasks to prompt LLM-based systems to produce domain-informed explanations. Those works were a significant source of inspiration for us. An et al. [1], proposed a logic-based explanation framework for Monte Carlo Tree Search, addressing its interpretability challenges. The framework translates user queries into logic and variable statements to ensure consistency with the underlying Markov Decision Process. LeMasurier et al. [22], discussed a framework that to produce explanations using LLMs that are fed with the robot’s behavior trees using the intermediate format proposed in [19]. Retrieval approaches leveraging robot logs [39] have been proposed to manage the limitations of long contexts, and have been further strengthened by incorporating a Visual Language Model (VLM) [40]. In other works, researchers use episodic information from robot past experiences in consonance with LLMs to generate explanations of behaviors, decisions and failures [3, 8, 23]. Finally, Bustamante et al. [7], combine robots’ internal knowledge coming from different robot modules (task model, world model, episodic memory, etc.) with LLMs, to answer questions about robots’ knowledge. Our work might fit in their framework, since our ontology-based robot knowledge may be one of their robot modules.

### 3 BLUEPRINT FOR ROBOT ONTOLOGY-BASED EXPLANATIONS

#### 3.1 Background

**Background on explainable agency:** The literature on explainable agency offers a well-established functional roadmap for designing explainable agents around four core functionalities [21]: (1) evaluating different candidate solutions to a task, (2) systematic indexing and storage of evaluation insights, (3) retrieval of stored knowledge, and (4) transmission of retrieved knowledge. First, during problem solving, the agent must evaluate multiple candidate solutions, analyzing their properties, so that it is possible to discriminate between them. Second, the agent must systematically index and store detailed records of its evaluation and classification in a structured repository, such as an episodic memory. Third, the agent must transform queries into retrieval cues to access relevant information from its memory. Fourth, once the information is retrieved, the agent translates it into human-interpretable language.

Inspired by these ideas, we propose a methodology that tailors these four functionalities or steps to the semantic richness of human-robot interactions (see Fig 1). Our proposal comprises: (1) robot experiences analysis and computation of their individual plan’s properties, (2) ontology-based episodic storage and inference for plan comparison and classification, (3) selective and systematic retrieval of knowledge-based narratives, and (4) LLM-based generation of the final explanation for its transmission.

**Insights from the social sciences:** There exists extensive research in philosophy, psychology, and cognitive science investigating how people define, generate, select, evaluate, and present explanations, showing that cognitive biases and social expectations shape the explanation process. Miller [26] argues that the field of explainable artificial intelligence (XAI) can build on this existing research. Miller reviewed key works from philosophy, cognitive psychology/science, and social psychology, highlighting major findings and discussing how they can be integrated into XAI research and practice. The most important findings state that explanations are *contrastive*, *selected*, and *social*. They are contrastive because they respond to counterfactual questions such as why one plan is better than another. Explanations are selective, typically presenting only part of the reasons, drawn from broader knowledge according to specific criteria. Finally, explanations are social, transferring knowledge in conversational form as part of interactions between agents. These qualities are inherent in the nature of the explanations themselves. Hence, being contrastive, selected and social are not just desirable but essential for the design and development of explainable robots. Note that there is a relationship between these properties and the functionalities (steps) discussed earlier. Contrastive explanations require the analysis of the qualities of multiple alternative options (functionality 1). The (biased) selection of explanation content depends on the knowledge storage and retrieval (functionalities 2 and 3), and can be learned from users’ preferences. Finally, social explanations demand transmission in a human-interpretable form (functionality 4).

Inspired by these ideas, this work proposes a methodology to produce *contrastive* and *social* (interactive) explanations that are *selected* according to different levels of specificity or detail.

#### 3.2 Experiential plans’ properties extraction

In the first step of the proposed methodology, each plan derived from collaborative experiences between humans and robots is analyzed, focusing on finding the plans’ properties. Note that these experiences contain variability and comprise different alternatives to perform the collaborative task, since humans do not always behave in a repetitive manner. Hence, the experiences are individually studied, extracting and computing the properties of the different executed plans. We specifically focused on three main aspects of plans: cost, makespan, and number of tasks. We are using the same plan qualities defined in a literature work that is later used as the evaluation baseline [32]. We note that the approach allows us to consider other qualities, e.g. the workload distribution between human and robot. However, for our purpose, it is enough to stick to these. Note that the actual computation of some of the properties is application-dependent, for instance, the cost might be related to different aspects: energy consumption, the existence in the plan of complex or unsafe tasks, etc. In our case, cost was related to how much human intervention was needed (see Sec. 4.2 for more details).

#### 3.3 Plans’ knowledge inference and storage

In the second step of our methodology, robots use the semantic structure of an ontology to infer knowledge about the plans’ properties comparison and classification, and store it in a knowledge-based episodic memory (i.e., a time-indexed knowledge base).

**3.3.1 Comparing plans’ properties and plans.** The ontology for modeling the properties of plans defined in [32], is grounded using the information extracted in the previous step. Then, a set of logic rules, also defined in that work, is used to infer the comparison of the different experiences: e.g., plans with larger cost are defined as more expensive than other ones with lower cost.

**3.3.2 Classifying plans’ properties and plans.** We intend to further reflect on the extracted data, analyzing the properties of the different plans, and finding patterns that are useful for the robot to classify experiences (e.g. a subset of plans is more recurring than others). Specifically, we propose to classify the different properties of plans as typical or atypical depending on whether they are within the robot experiences’ mode or not. One might argue that this can be done by predefining a threshold of typical values for each property. However, that would require prior knowledge about the properties’ common values, hindering the flexibility of the approach. Instead, we prefer to automatically extract this threshold from the actual data using a well-known state-of-the-art method for this: the Highest Density Interval (HDI). In this case, the assumption is the availability of a set of experiences. HDI is the narrowest interval within a probability distribution that captures a given probability mass (or sample proportion), representing the most recurring values of a parameter. Unlike equal-tailed intervals, the HDI is a Bayesian measure that always includes the mode and ensures that all values inside the interval have a higher probability density than those outside. This makes it a more direct and intuitive way to express uncertainty in the value of a parameter.

**A novel implementation for computing HDI:** The HDI is widely used in Bayesian analysis and robust descriptive statistics because it always contains the region of highest density. However,

---

**Algorithm 1:** Empirical HDI via Sliding Window

---

**Input:** Sample  $X = \{x_1, \dots, x_n\}$ , coverage  $\alpha \in (0, 1)$   
**Output:** Empirical HDI interval  $[\text{lo}, \text{hi}]$

- 1 Sort  $X$  into  $X_{(1)} \leq X_{(2)} \leq \dots \leq X_{(n)}$
- 2  $k \leftarrow \lceil \alpha \cdot n \rceil$
- 3  $\text{best\_width} \leftarrow \infty, \text{best\_start} \leftarrow 0$
- 4 **foreach**  $i \in (1, n - k + 1)$  **do**
- 5      $\text{width} \leftarrow X_{(i+k-1)} - X_{(i)}$
- 6     **if**  $\text{width} < \text{best\_width}$  **then**
- 7          $\text{best\_width} \leftarrow \text{width}$
- 8          $\text{best\_start} \leftarrow i$
- 9     **end**
- 10 **end**
- 11  $\text{lo} \leftarrow X_{(\text{best\_start})}$
- 12  $\text{hi} \leftarrow X_{(\text{best\_start}+k-1)}$

---

its computation requires to know the probability distribution of the robot’s dataset. For this, it is usual to compute the empirical HDI, a non-parametric approximation based on sorted data, requiring no distributional assumptions.

Given a sample  $\{x_1, \dots, x_n\}$ , containing the values of a property for  $n$  plans, and a coverage (mass)  $\alpha \in (0, 1)$ , the empirical HDI approximates the theoretical HDI by finding the shortest interval containing at least  $\alpha n$  points:

$$k = \lceil \alpha \cdot n \rceil, \quad \widehat{I}_\alpha = [x_{(i^*)}, x_{(i^*+k-1)}],$$

where

$$i^* = \arg \min_{1 \leq i \leq n-k+1} (x_{(i+k-1)} - x_{(i)}).$$

Alg. 1 shows the pseudo-algorithm that we propose to compute the empirical HDI using a sliding window over a set of  $n$  values, where  $n$  is the number of plans (experiences). Note that since HDI is univariate, this method must be applied  $m$  times (the number of plan’s properties) to find the interval for each property (e.g., the number of tasks, cost, etc.). Then, every specific plan’s property is classified as typical or atypical depending on whether its value is within the interval or not. To analyze the computational complexity of the algorithm, we pay attention to its parts. The empirical HDI algorithm consists of sorting the sample of size  $n$ :  $O(n \log n)$ , scanning with a sliding window:  $O(n)$ . Hence, the overall time complexity is:

$$O(n \log n), \quad \text{with space complexity } O(n).$$

**A new ontological model for typical plan classification:**

The content related to whether a plan’s property is typical or atypical is newly introduced in our work; thus it is necessary a novel set of ontological entities and rules for knowledge representation and reasoning. The intended model shall cover a new ontological scope: *how can a plan and its qualities be classified (e.g. typical)?*.

The new model builds upon the ontology introduced in [32], extending and reusing its existing structure. Consequently, it inherits its upper ontology, the DOLCE+DnS Ultralite (DUL) foundational ontology [6]. In DUL, the way to classify an Entity is through a Concept, thus we propose new sub-classes of Concept: Typical Plan Quality Value, Atypical Plan Quality Value, Typical

Plan, and Atypical Plan. The first two are used to classify plan qualities (e.g. Plan Cost), while the two remaining classify Plans. The four new ontological classes are formalized using the description logic version of the Web Ontology Language 2, OWL 2 DL. The new model is instantiated by grounding the result of the HDI computation on each of the properties of interest: number of tasks, makespan, and cost.

**Logical rule to infer whether a plan is typical or atypical:**

Assume a consistent instantiated ontology  $O_i$  as a set of triples  $\langle \text{subject}, \text{predicate}, \text{object} \rangle$ , which encodes knowledge about the qualities of different plans and the classification of the qualities as typical or atypical. The next step is to infer which plans are typical. Here, the criterion to decide if a plan is typical is satisfied when all its qualities are typical (i.e. they have a typical value). This is, if every quality of a plan ( $D$ ) is classified by Typical Plan Quality Value:

$$\forall Q \exists D \langle Q, \text{rdf.type}, \text{dul.Quality} \rangle \wedge \langle D, \text{rdf.type}, \text{dul.Plan} \rangle \wedge \langle Q, \text{dul.isQualityOf}, D \rangle \wedge \langle Q, \text{dul.isClassifyBy}, \text{TypicalPlanQualityValue} \rangle$$

The previous logical expression reads as follows: all qualities ( $Q$ ) related to a plan ( $D$ ), ‘are classified by’ *TypicalPlanQualityValue*. The predicate ‘*rdf:type*’, which relates an individual with its class, is a standard predicate from the Resource Description Framework (RDF) [25]. When the previous logical expression is true, we can say that  $D$  ‘is classified by’ *TypicalPlan*, and the knowledge indicating it would be asserted:

$$O_i \leftarrow O_i \cup \langle D, \text{dul.isClassifiedBy}, \text{TypicalPlan} \rangle.$$

In contrast, when at least one of the properties of a plan is atypical, the plan  $D$  will be classified as atypical:

$$O_i \leftarrow O_i \cup \langle D, \text{dul.isClassifiedBy}, \text{AtypicalPlan} \rangle.$$

These logical expressions were combined into a single rule to classify a plan as typical or atypical. The robot knowledge base used in this work is written in Prolog [11], thus the rule was implemented as a Prolog predicate and is accessible online.<sup>1</sup> The time complexity of our implementation is  $O(1)$  (constant complexity) in the best case, and  $O(m)$  in the worst case, where  $m$  is the number of qualities of a plan. The space complexity is also independent of the input size  $O(1)$ . In practice, this rule is run for multiple plans, thus being  $n$  the number of plans, the overall time complexity is:  $O(n)$  for the best case and  $O(n \cdot m)$  the worst case. The overall space complexity remains  $O(1)$ , regardless of the number of plans  $n$  or qualities  $m$ . Note that the worst case will occur only when all the qualities of all the plans are typical, which is unlikely in our scenario, since HDI ensures that only a subset of qualities are classified as typical.

### 3.4 Selective explanatory knowledge retrieval

The third step of our methodology deals with selecting two plans and retrieving the explanation content according to a specific criterion. Our method focuses on contrastive knowledge (comparing the two plans), and allows specifying the level of detail. For this, our methodology integrates a literature method, ACXON [32], an algorithm designed to retrieve knowledge about divergences between ontological entities (e.g., plans) and to generate textual contrastive

<sup>1</sup><https://github.com/indigo-the-agent/know-demo>

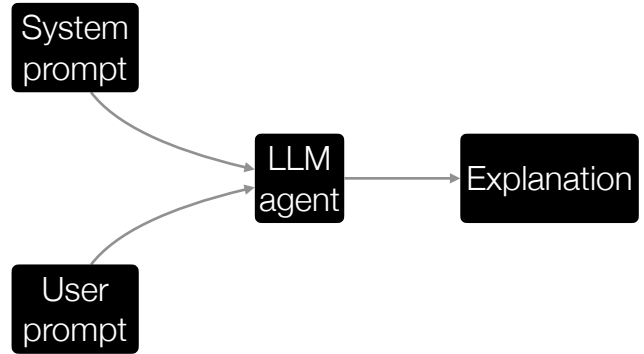
explanatory narratives. It operates on four main inputs: a time-indexed knowledge graph (triples annotated with a time interval in which they hold), the ontological class(es) of the instances to narrate, the temporal locality (interval of interest), and the desired level of specificity. Although it was conceived to work with contrastive narratives about pairs of robot plans, ACXON is general enough to be applied to other ontologies and entity classes. Indeed, because it leverages the structure of the knowledge graph and the vicinity between typical entities, it can automatically retrieve the information about typical properties and plans that we have introduced in our work. Depending on the chosen specificity level, ACXON produces three types of contrastive narratives, where specificity determines the granularity of detail included in the narrative i.e., the number of knowledge tuples. The retrieved knowledge tuples are aggregated into a final textual narrative, using some rules commonly used by humans when asked to combine chunks of knowledge. Note that narratives only contain sound ontological knowledge extracted from the knowledge base plus some basic connectors such as ‘and’, or ‘while’. The textual format of the retrieved knowledge fits perfectly our proposed architecture in which the knowledge is fed to an LLM agent. For a set of  $n$  robot plans (experiences), a total of  $\frac{n(n-1)}{2}$  different pairwise narratives could be retrieved in this step. In the following, we present an example narrative of specificity 3 that compares a pair of plans (X and Y).

**Ontology-based narrative**

*‘Plan X’ is cheaper plan than and is shorter plan than and is faster plan than and is better plan than ‘Plan Y’. ‘Plan X’ has makespan ‘Plan X makespan’; while ‘Plan Y’ has makespan ‘Plan Y makespan’. ‘Plan X’ has number of tasks ‘Plan X number of tasks’; while ‘Plan Y’ has number of tasks ‘Plan Y number of tasks’. ‘Plan X’ has cost ‘Plan X cost’; while ‘Plan Y’ has cost ‘Plan Y cost’. ‘Plan X’ is classified by ‘TypicalPlan’; while ‘Plan Y’ is classified by ‘AtypicalPlan’. ‘Plan X makespan’ has value ‘28.20’; while ‘Plan Y makespan’ has value ‘40.35’. ‘Plan X number of tasks’ has value ‘12’; while ‘Plan Y number of tasks’ has value ‘18’. ‘Plan X cost’ has value ‘0’; while ‘Plan Y cost’ has value ‘3’.*

### 3.5 Robot explanation generation

In the fourth and last step of the proposed methodology, an LLM agent uses the retrieved narrative and translates it into a more human-interpretable format to be transmitted. Recall that the intended explanation shall be sound and natural, which we translate into three main explanation qualities: brevity, clarity, and semantic coherence. There might be many prompting strategies to produce the intended explanation, but exploring them would be an engineering effort that is out of the scope of our scientific work. Hence, we decided to start the agent with a simple prompt that is as follows:



**Figure 2: LLM-based explanation generation procedure: a single system prompt and a user prompt (ontology-based narrative) are used to generate the explanation.**

**System prompt sent to the LLM**

You are an agent that based on a given ontology-based narrative, shall provide a new narrative that: (a) is shorter than the original, (b) uses an easier language than the original, and (c) keeps the semantic meaning of the original.

This prompt is sent to the agent as a system prompt, a set of initial instructions that provide the language model with context, defines its role, and establishes its behavioral guidelines for an interaction. Then, an ontology-based narrative comparing a pair of experiences (plans) is fed as a user prompt, so that the LLM agent returns a refined explanation. This process is repeated for each narrative comparing a pair of plans retrieved in the previous methodology’s step, for  $n$  plans, that would be one of the  $\frac{n(n-1)}{2}$  narratives. The implementation<sup>2</sup> provides an environment that is later used to evaluate the approach (see Fig. 2), a setting where the agent provides an explanation given a system and a user prompt.

The LLM-based agent was deployed using *Ollama*,<sup>3</sup> an open-source platform optimized for running large language models locally. Additionally, we employed *Pydantic AI*,<sup>4</sup> a Python-based agent framework designed for developing production-grade applications and workflows powered by generative AI.

## 4 EVALUATION OF ROBOT ONTOLOGY-BASED EXPLANATIONS

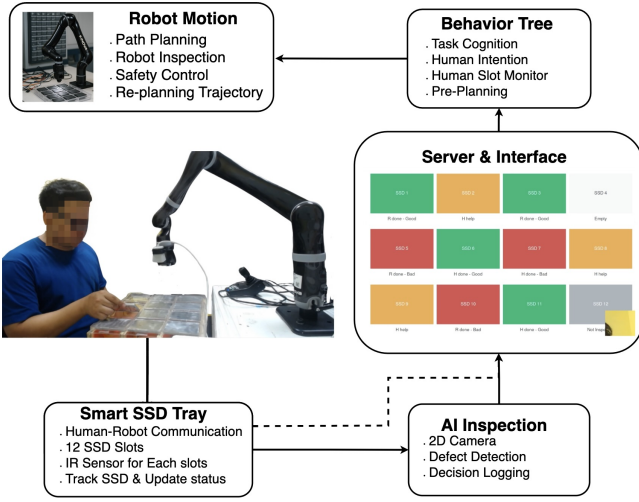
### 4.1 Collaborative SSD case inspection

The evaluation of the proposed methodology is contextualized in a lab mock-up of a real industrial task, a human-robot collaborative inspection of the surface of metallic SSD cases. In this inspection task, there is a tray with 12 SSD units, and the robot helps the operator inspecting some of the units, although the robot might also try to perform the entire task. The robot is able to recognize when the precision of its inspection is low for a particular piece and ask the human for reassurance. The setup combines a 6-DOF

<sup>2</sup><https://github.com/indigo-the-agent/know-rox>

<sup>3</sup><https://ollama.com/>

<sup>4</sup><https://ai.pydantic.dev/>



**Figure 3: Overview of the implementation for the collaborative SSD inspection task.**

robotic arm with a 2D camera for visual data capture, an AI-based defect detection model for real-time classification, and a user interface for efficient human-robot interaction. The implementation of the task comprises four core modules: a robotic manipulator for motion execution and trajectory planning, an AI-driven visual inspection module, a sensor-equipped smart SSD tray, and a behavior tree-based task execution model. Together, these modules enable synchronous collaboration between the human and the robot. A server-side graphical user interface (GUI) offers real-time visualization of task progress and inspection results. Note that communication from the robot to the human to indicate which cases need a second inspection occurs through the GUI. The complete implementation pipeline is shown in Fig. 3.

## 4.2 Evaluation procedure and setup

**4.2.1 Generating human-robot collaborative experiences.** In total, we collected 18 executions of the task, completing the inspection of 12 SSD cases, with a single human. To generate variability, we varied the confidence threshold of the robot’s visual inspection and also the sequence of actions that the human followed. When the robot was uncertain whether one SSD case was damaged or not, i.e. classification confidence below the threshold, the case was left for the human to review. Data for each experience were collected using *rosbags*, a file format and tool within the Robot Operating System (ROS) used to store ROS message data. The data includes the tray state during the whole task (to be inspected, damaged, correct), the sequence of actions of the human and the robot, and images from the robot’s visual inspection.

**4.2.2 Experiential plans’ properties extraction.** We manipulated the data from the collaborative SSD case inspection scenario, and extracted the knowledge that we considered relevant for describing the experiences: the number of tasks, the makespan, and the cost. The number of tasks includes all the SSD cases inspected by the human and the robot, which may be twelve or more, since the cases in

which the robot is not confident are inspected twice. The makespan is calculated as the total time that it takes to complete the inspection of the entire tray. Finally, the cost is equal to the number of times that the robot doubted. HDI was computed for each property using a mass value of 0.68, which means that at least 68% of the plans’ properties are classified as typical. This coverage or mass is often called *one-sigma*, since given a data set approximately distributed normally, approximately 68% of the data is within one standard deviation of the mean (according to the *empirical rule*).

**4.2.3 Plans’ knowledge inference and storage.** We initialized a knowledge base, which works as an episodic memory, and asserted the abstraction of the data obtained in the previous step in the form of ontological triples. First, the knowledge about the comparison between plans’ properties and plans. This includes knowledge such as:

$\langle \text{PlanX Cost, hasBetterQualityValueThan, PlanY Cost} \rangle$ .

We also asserted knowledge of whether the plan’s qualities were typical or not:

$\langle \text{PlanX Cost, dul.isClassifyBy, TypicalPlanQualityValue} \rangle$ .

Once the knowledge was stored in the ontology-based episodic memory, we run the logical rules to infer how the different plans relate to each other, and also to classify plans as typical or atypical. The rules produced new knowledge, which was also asserted to the knowledge base:  $\langle \text{PlanX, ocr.isMoreExpensivePlanThan, PlanY} \rangle$ , and  $\langle \text{PlanX, dul.isClassifiedBy, TypicalPlan} \rangle$ .

**4.2.4 Selective explanatory knowledge retrieval.** Once there was an active knowledge base containing all the knowledge about how the different experiences or plans compare and are classified, we retrieved the content to be used in the explanations. This was generated using the state-of-the-art algorithm for ontology-based narrative generation [32]. This algorithm takes four inputs: the time-indexed knowledge graph (the active knowledge), the ontological class(es) of the instances to compare is (`dul.Plan`), the temporal locality (a time interval  $(_, \text{Inf})$ ), and the desired level of specificity (the three levels are used). Hence, we retrieve  $\frac{n(n-1)}{2}$  narratives comparing in pairs the 18 plans ( $n = 18$ ) between each other, in total 153 narratives.

**4.2.5 Robot explanation generation.** The retrieved narratives are sent to the LLM agent, which generates individual explanations using each of them. We used the *gpt-oss:20b* model, a 20 billion-parameter version of openAI’s open-weight models designed for powerful reasoning, agentic tasks, and versatile developer use cases. We decided to use this model because it is openly available and because it provides a great trade-off between model size and accuracy. It was important for us that the model could be easily run in a single GPU since in the future we would like to be able to run our approach in real robot execution time. Note that other models can be easily integrated into our methodology.

**4.2.6 Computer setup for running the experiments.** The software for the test was run on a desktop PC with an Intel Core i9-11900KF 11th Gen CPU (8x 3.50 GHz), 64 GB DDR4 RAM, and an NVIDIA GeForce RTX 3090 GPU.

### 4.3 Evaluation metrics

The evaluation was conducted using a set of objective metrics, selected to align with established practices in the state-of-the-art. The metrics aim to evaluate the three main intended features of our generated explanations: brevity, clarity, and semantic coherence.

**Explanation length.** This metric was used to evaluate the brevity of the explanations. In this work, the metric is calculated as the number of words  $N_w$ . Explanations are social, thus a good quality index is to measure how long they are, which is proportional to how much time it would take to communicate them.

**Readability score.** To evaluate the clarity of the explanations, we used the Flesch Reading Ease (FRES) readability  $\mathcal{R}_{fres}$  metric [20], proposed in the 1970s and still one of the best tools to evaluate how easy something is to read. The FRES test measures readability by analyzing word, syllable, and sentence counts. It computes the average number of words per sentence and syllables per word, assuming that shorter words and sentences enhance readability. Higher scores indicate easier-to-understand text. The maximum possible FRES score is 121.22, achieved when every sentence consists of a single one-syllable word. There is no defined lower bound, thus highly complex sentences may produce negative scores. We used the implementation of FRES from the Textstat Python library.<sup>5</sup> Other readability metrics were considered and tried: automated readability index, Gunning fog index, Dale-Chall score [10], etc. However, results were similar, so we decided to use FRES because it is one of the most well-known metrics in the state-of-the-art.

**Semantic similarity.** This metric is used to evaluate whether the process of transforming the initial ontology-based narrative into a shorter and clearer explanation results in a loss of semantic soundness. We used the cosine similarity  $\mathcal{S}_{cs}$  function from the *Scikit-learn* Python library,<sup>6</sup> which directly calculates the cosine similarity between two vectors or matrices, returning a value  $\in (0, 1)$ . The retrieved ontology-based narrative and the explanation generated with the LLM agent were converted into a matrix of token counts using *CountVectorizer* from the same library. In this process, we avoided counting most common words in English text, such as ‘and’, ‘the’, etc., which are presumed to be uninformative in representing the content of the text and which were removed to avoid them being used in measuring similarity between the two texts.

### 4.4 Results of evaluating the brevity, clarity and semantic coherence of the explanations

A statistical analysis was conducted to evaluate the significance of the improvements achieved by the proposed approach in terms of brevity and clarity with respect to the baseline, while preserving the semantic coherence of the ontology-based narratives. The two different methods were compared across three levels of specificity (independent variable) and assessed using the metrics introduced in Section 4.3 (dependent variables). The normality of the difference between the results obtained by both methods was examined using the Shapiro-Wilk test ( $\alpha = 0.05$ ). For all cases, the normality assumption can be rejected ( $p < 0.05$ ), but since the distribution of the difference between the results was approximately symmetric

**Table 1: Average evaluation results for each metric and the 153 pairs of plans. The mean difference is statistically significant  $p < 0.001$  for all the cases, with the only exception of  $\mathcal{R}_{fres}$  for specificity 1.**

Method	Baseline (ACXON)			Ours		
	1	2	3	1	2	3
$N_w$	28	741.7	1788.6	<b>18.7</b>	<b>104.7</b>	<b>131.8</b>
$\mathcal{R}_{fres}$	<b>87</b>	67.6	48.2	86.3	<b>80.7</b>	<b>84.6</b>
$\mathcal{S}_{cs}$	-	-	-	<b>0.7107</b>	<b>0.8668</b>	<b>0.7935</b>

**Table 2: Statistical evaluation results for each metric and specificity level. The specific statistical test metrics for each case is denoted between square brackets.**

Specificity	1	2	3
$N_w$ [ $t(152), p$ ]	[31.1, < .001]	[16.7, < .001]	[14.7, < .001]
$\mathcal{R}_{fres}$ [ $t(152), p$ ]	[1.9, .062]	[8.4, < .001]	[15.4, < .001]
$\mathcal{S}_{cs}$ [ $t(152), p$ ]	[9.9, < .001]	[31.8, < .001]	[15.3, < .001]

and the number of samples was sufficiently large ( $N=153$ ), parametric tests can be used. *Paired-t test* was used for the **explanation length** and the **readability score**, while *one-sample-t test* was applied for the **semantic similarity** metric.

For the readability score at specificity 1, no significant difference was observed between the baseline and the proposed approach, although both yielded high readability scores. This outcome was expected, as baseline ontology-based narratives at this level are typically short and inherently easy to read. For the rest of the metrics and levels of specificity, the statistical results indicate that there is a significant large difference between the baseline and our method  $p < 0.001$ . The average results of the evaluation metrics for the 153 pairs of plans, across both approaches and specificity levels, are summarized in Tab. 1. The detailed statistical analysis is reported Tab. 2.

The performance of the proposed approach is consistently superior to that of the baseline ontology-based narratives, especially for the second and third levels of specificity. The proposed explanations are substantially shorter and clearer, while maintaining high semantic fidelity. Specifically, considering **explanation length**, our explanations are a 33%, 86% and 93% shorter on average than the baseline explanations for the specificity level 1, 2, and 3, respectively. In the case of the **readability score**, the proposed method produces explanations that are a 19% and 76% easier to read and comprehend than the baseline narratives at levels 2 and 3. Importantly, this considerable gain in brevity and clarity does not come at the cost of **semantic coherence**. Indeed, the semantic cosine similarity was consistently above 0.7, indicating a strong semantic correlation between the generated explanations and the baseline.

These results demonstrate that integrating ontological reasoning with LLM-based refinement enables the generation of concise yet semantically faithful explanations. The findings highlight the complementary roles of symbolic and neural representations: while the ontology guarantees logical consistency and grounding, the LLM enhances linguistic fluidity and contextual adaptation.

<sup>5</sup><https://pypi.org/project/textstat/>

<sup>6</sup><https://scikit-learn.org/>

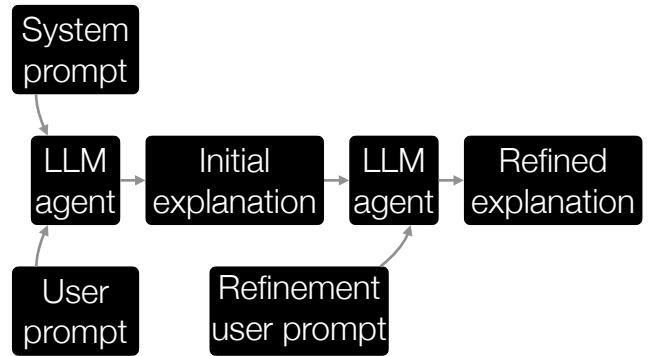
## 4.5 Qualitative evaluation and discussion

The quantitative evaluation discussed earlier in this article offers a rigorous objective analysis of the strengths of our proposed methodology. However, there are some aspects of the explanations that require a more qualitative study, thus, this section discusses them.

**4.5.1 Practical view on the brevity improvement.** The explanations discussed in this work would often be transmitted from robots to humans, who would need to understand and retain the explanatory content. Since our explanations are textual, humans would either listen to or read them. From state-of-the-art studies we know that to retain information, people are comfortable with a speaking pace of 150-160 words per minute (wpm), while the pace for silent reading is 250-400 wpm [35]. Let’s consider the lowest bound of those intervals, and the average explanation lengths reported in Sec. 4.4. For the baseline narratives, humans would spend almost **5 minutes** listening to an average narrative of specificity 2, and **12 minutes** for one of specificity 3. Our method reduces those times to **42 and 55 seconds**, respectively. In the case of silent human reading, the baseline narratives of specificity 2 and 3 would require **3 and 7 minutes**, while our explanations only **25 and 32 seconds**. This is definitely a huge advancement towards the broader goal of creating artificial agents capable of reasoning and communicating in ways that are both technically rigorous and socially acceptable, bridging the gap between formal knowledge and human-centric expression.

**4.5.2 A close look at unexpected explanations.** We visually inspected the explanations generated using our approach, reading them looking for major unexpected formats or content. Our produced explanation is often a single paragraph much shorter and more clearly stated than the original ontology-based narrative. This is consistent with the objective quantitative results. However, in a few cases the explanations contain a brief summary comparing the pair of plans and some bullet points with more direct and concise major differences between the plans. This only occurs in explanations of specificity levels 2 and 3, being a bit more usual in level 3. In those cases, the explanations are still shorter and clearer than their respective ontology-based narratives. However, we found one case in which the generated explanation is really difficult to read and unusually long: *Pair 27* for specificity level 3. In this case, the original ontological narrative contains 4876 words and has a readability score of -18.42. The explanation generated with our approach contains 941 words, a bad result but still shorter, and a readability score of -147.33, which is much worse than the baseline score. Our explanation includes multiple bullet points, detailing, one by one, almost all the specific differences between the pair of plans. Note that this is clearly an outlier, in fact, the second lowest readability score of our explanations is 56.44 (*Pair 136*). As a major takeaway of this qualitative visual inspection, we would recommend potential users of our methodology to inspect the explanations generated for their use cases, and refine the prompts to avoid undesirable behaviors.

**4.5.3 Discussion on an extension to interactive scenarios.** Blending the use of ontologies with the use of LLMs, our methodology has proven to effectively produce sound and natural explanations, improving the brevity and clarity of ontology-based narratives. Furthermore, the use of ontology-informed LLMs also brings a powerful interactive capability that enhances robot explainability



**Figure 4: LLM-based explanation generation as an interactive procedure: Once the initial explanation is generated, a second user prompt is used to refine the explanation (e.g., asking for a shorter or clearer version).**

by enabling fluid, context-aware dialogue with humans. Unlike static explanation systems, LLMs can dynamically adapt their responses to user queries, clarify ambiguities, and tailor explanations based on conversational feedback. This interactivity allows robots to provide not only one-way justifications but also engage in explanatory dialogue—answering follow-up questions, simplifying complex reasoning, or expanding on technical details when requested. In our case, after receiving an initial explanation, users might require shorter or clearer versions. They might also ask for specific details that were skipped when shortening the ontology-based narrative: e.g., the actual value of the cost of the two plans, not just if one is more expensive. For these cases, our methodology can be easily extended by slightly modifying the fourth and last step. Fig. 4 depicts an overview of the proposed modification.

## 5 CONCLUSION

This work demonstrated the potential of integrating ontology-based reasoning with large language models (LLMs) to advance explainable human-robot interaction. By combining the logical rigor and semantic grounding of ontologies with the natural, adaptive communication abilities of LLMs, our framework enables robots to generate explanations that are both accurate and accessible to humans. The experimental validation in an industrial collaborative setting showed that this hybrid approach significantly improves the clarity and conciseness of ontology-based narratives without sacrificing their semantic depth. These results underscore the value of uniting structured symbolic knowledge with data-driven language generation for building transparent, trustworthy, and socially aware robotic systems. Future research will focus on extending this integration toward **real-time dialogue**, **multimodal explanations**, and **experiments with users** to assess the impact of such explanations on subjective perceptions of trust and collaboration. Additionally, we aim to explore how these explanation mechanisms can be used not only for human understanding but also for robot self-assessment, enabling robots to transform explanations from merely communicative tools into mechanisms for adaptive reasoning and self-improvement in **long-horizon tasks**.

## REFERENCES

- [1] Ziyang An, Xia Wang, Hendrik Baier, Zirong Chen, Abhishek Dubey, Taylor T. Johnson, Jonathan Sprinkle, Ayan Mukhopadhyay, and Meiyi Ma. 2025. Combining LLMs with a Logic-Based Framework to Explain MCTS. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems* (Detroit, MI, USA) (AAMAS '25). International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 2405–2407.
- [2] Sule Anjomshoae, Amro Najjar, Davide Calvaresi, and Kary Främling. 2019. Explainable Agents and Robots: Results from a Systematic Literature Review. In *Proceedings of the 18th International Conference on Autonomous Agents and Multi-Agent Systems* (Montreal QC, Canada) (AAMAS '19). International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 1078–1088.
- [3] Abrar Anwar, John Welsh, Joydeep Biswas, Soha Pouya, and Yan Chang. 2025. Remembr: Building and reasoning over long-horizon spatio-temporal memory for robot navigation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2838–2845.
- [4] Daniel Beßler, Robert Porzel, Mihai Pomarlan, Michael Beetz, Rainer Malaka, and John Bateman. 2020. A formal model of affordances for flexible robotic task execution. In *Proceedings of the 24th European Conference on Artificial Intelligence (ECAI 2020)*. IOS Press, 2425–2432.
- [5] Stefano Borgo, Amedeo Cesta, Andrea Orlandini, and Alessandro Umbrico. 2019. Knowledge-based adaptive agents for manufacturing domains. *Engineering with Computers* 35, 3 (01 Jul 2019), 755–779.
- [6] Stefano Borgo, Roberta Ferrario, Aldo Gangemi, Nicola Guarino, Claudio Masolo, Daniele Porello, Emilio M. Sanfilippo, and Laure Vieu. 2022. DOLCE: A descriptive ontology for linguistic and cognitive engineering. *Applied Ontology* 17, 1 (2022), 45–69.
- [7] Samuel Bustamante, Markus Knauer, Jeremias Thun, Stefan Schneyer, Alin Albu-Schäffer, Bernhard Weber, and Freek Stulp. 2025. RACCOON: Grounding Embodied Question-Answering with State Summaries from Existing Robot Modules. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 4322–4329.
- [8] Leonard Bärmann, Chad DeChant, Joana Plewnia, Fabian Peller-Konrad, Daniel Bauer, Tamim Asfour, and Alex Waibel. 2025. Episodic Memory Verbalization using Hierarchical Representations of Life-Long Robot Experience. arXiv:2409.17702 [cs.RO] <https://arxiv.org/abs/2409.17702>
- [9] Gerard Canal, Senka Krivić, Paul Luff, and Andrew Coles. 2022. PlanVerb: Domain-Independent Verbalization and Summary of Task Plans. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36.
- [10] J.S. Chall and E. Dale. 1995. *Readability Revisited: The New Dale-Chall Readability Formula*. Brookline Books. <https://books.google.es/books?id=2nbuAAAAMAAJ>
- [11] William F. Clocksin and Christopher S. Mellish. 2012. *Programming in Prolog: Using the ISO standard*. Springer-Verlag Berlin Heidelberg.
- [12] Tulving Endel. 1972. Episodic and semantic memory. In *Organization of memory*, E. Tulving and W. Donaldson (Eds.). Academic Press, New York, 381–403.
- [13] Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A Survey on RAG Meeting LLMs: Towards Retrieval-Augmented Large Language Models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, 6491–6501.
- [14] S. R. Fiorini, J. Bermejo-Alonso, P. Gonçalves, E. Pignaton de Freitas, A. Olivares-Alarcos, J. I. Olszewska, E. Prestes, C. Schlenoff, S. V. Ragavan, S. Redfield, B. Spencer, and H. Li. 2017. A Suite of Ontologies for Robotics and Automation [Industrial Activities]. *IEEE Robotics Automation Magazine* 24, 1 (2017), 8–11.
- [15] Jorge Garcia Flores, Iván Meza, Émilie Colin, Claire Gardent, Aldo Gangemi, and Luis A. Pineda. 2018. Robot experience stories: First person generation of robotic task narratives in SitLog. *Journal of Intelligent & Fuzzy Systems* 34 (2018), 3291–3300. 5.
- [16] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-Augmented Generation for Large Language Models: A Survey. arXiv:2312.10997 [cs.CL]
- [17] Paulo J.S. Gonçalves, Alberto Olivares-Alarcos, Julita Bermejo-Alonso, Stefano Borgo, Mohammed Diab, Maki Habib, Hirenkumar Nakawala, S. Veera Ragavan, Ricardo Sanz, Elisa Tosello, and Howard Li. 2021. IEEE Standard for Autonomous Robotics Ontology [Standards]. *IEEE Robotics & Automation Magazine* 28, 3 (2021), 171–173.
- [18] Giancarlo Guizzardi and Nicola Guarino. 2024. Explanation, semantics, and ontology. *Data & Knowledge Engineering* 153 (2024), 102325.
- [19] Zhao Han, Daniel Giger, Jordan Allspaw, Michael S. Lee, Henny Admoni, and Holly A. Yanco. 2021. Building the Foundation of Robot Explanation Generation Using Behavior Trees. *J. Hum.-Robot Interact.* 10, 3, Article 26 (2021), 31 pages.
- [20] J Peter Kincaid, Robert P Fishburne Jr, Richard L Rogers, and Brad S Chissom. 1975. *Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel*. Technical Report.
- [21] Pat Langley. 2024. From Explainable to Justified Agency. In *Explainable Agency in Artificial Intelligence*, Silvia Tulli and David W. Aha (Eds.). CRC Press, 1–20.
- [22] Gregory LeMasurier, Christian Tagliamonte, Jacob Breen, Daniel Maccaline, and Holly A. Yanco. 2024. Templated vs. Generative: Explaining Robot Failures. In *2024 33rd IEEE International Conference on Robot and Human Interactive Communication (ROMAN)*. 1346–1353.
- [23] Zeyi Liu, Arpit Bahety, and Shuran Song. 2023. REFLECT: Summarizing Robot Experiences for Failure Explanation and Correction. arXiv:2306.15724 [cs.RO] <https://arxiv.org/abs/2306.15724>
- [24] Tamlin Love, Antonio Andriella, and Guillem Alenyà. 2025. Temporal Counterfactual Explanations of Behaviour Tree Decisions. arXiv:2509.07674 [cs.RO] <https://arxiv.org/abs/2509.07674>
- [25] Brian McBride. 2004. *The Resource Description Framework (RDF) and its Vocabulary Description Language RDFS*. Springer Berlin Heidelberg, Berlin, Heidelberg, 51–65.
- [26] Tim Miller. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence* 267 (2019), 1–38.
- [27] Yasamin Moghbelan, Alfonso Esposito, Ivan Zyrianoff, Giulia Spaletta, Stefano Borgo, Claudio Masolo, Fabiana Ballarin, Valeria Seidita, Roberto Toni, Fulvio Barbaro, Giusy Di Conza, Francesca Pia Quartulli, and Marco Di Felice. 2024. A Smart Motor Rehabilitation System Based on the Internet of Things and Humanoid Robotics. *Applied Systems* 14, 24 (2024).
- [28] Muhammad Raza Naqvi, Linda Elmhadi, Arkopaul Sarkar, Bernard Archimede, and Mohamed Hedi Karray. 2024. Survey on ontology-based explainable AI in manufacturing. *Journal of Intelligent Manufacturing* 35, 8 (01 Dec 2024), 3605–3627.
- [29] Alberto Olivares-Alarcos, Antonio Andriella, Sergi Foix, and Guillem Alenyà. 2023. Robot explanatory narratives of collaborative and adaptive experiences. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 11964–11971.
- [30] Alberto Olivares-Alarcos, Daniel Beßler, Alaa Khamis, Paulo Gonçalves, Maki K. Habib, Julita Bermejo-Alonso, Marcos Barreto, Mohammed Diab, Jan Rosell, João Quintas, Joanna Olszewska, Hirenkumar Nakawala, Edison Pignaton, Amelie Gyrard, Stefano Borgo, Guillem Alenyà, Michael Beetz, and Howard Li. 2019. A review and comparison of ontology-based approaches to robot autonomy. *The Knowledge Engineering Review* 34 (2019), e29.
- [31] Alberto Olivares-Alarcos, Sergi Foix, Júlia Borràs, Gerard Canal, and Guillem Alenyà. 2024. Ontological Modeling and Reasoning for Comparison and Contrastive Narration of Robot Plans. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2405–2407.
- [32] Alberto Olivares-Alarcos, Sergi Foix, Júlia Borràs, Gerard Canal, and Guillem Alenyà. 2025. Ontological foundations for contrastive explanatory narration of robot plans. arXiv:2509.22493 [cs.RO] <https://arxiv.org/abs/2509.22493>
- [33] Andrew Ponomarev and Anton Agafonov. 2024. Ontology-Based Explanations of Neural Networks: A User Perspective. In *Artificial Intelligence in HCI*, Helmut Degen and Stavroula Ntoa (Eds.). Springer Nature Switzerland, Cham, 264–276.
- [34] Francisco Ramos, Cristian O. Scrob, Andrés S. Vázquez, Raúl Fernández, and Alberto Olivares-Alarcos. 2018. Skill-Oriented Designer of Conceptual Robotic Structures. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 5679–5684.
- [35] Keith Rayner, Elizabeth R. Schotter, Michael E. J. Masson, Mary C. Potter, and Rebecca Treiman. 2016. So Much to Read, So Little Time: How Do We Read, and Can Speed Reading Help? *Psychological Science in the Public Interest* 17, 1 (2016), 4–34.
- [36] Stephanie Rosenthal, Sai P Selvaraj, and Manuela M Veloso. 2016. Verbalization: Narration of Autonomous Robot Experience.. In *IJCAI*, Vol. 16. 862–868.
- [37] C. Schlenoff, E. Prestes, R. Madhavan, P. Gonçalves, H. Li, S. Balakirsky, T. Kramer, and E. Migueláñez. 2012. An IEEE standard Ontology for Robotics and Automation. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 1337–1342.
- [38] Alexander Smirnov and Andrew Ponomarev. 2023. Ontology-Based Explanations of Neural Networks for Collaborative Human-AI Decision Support Systems. In *Proceedings of the Seventh International Scientific Conference "Intelligent Information Technologies for Industry" (ITI'23)*, Sergey Kovalev, Igor Kottenko, and Andrey Sukhanov (Eds.). Springer Nature Switzerland, Cham, 353–362.
- [39] David Sobrin-Hidalgo, Miguel A. González-Santamarta, Ángel M. Guerrero-Higuera, Francisco J. Rodríguez-Lera, and Vicente Matellán-Olivera. 2024. Explaining Autonomy: Enhancing Human-Robot Interaction through Explanation Generation with Large Language Models. arXiv:2402.04206 [cs.RO] <https://arxiv.org/abs/2402.04206>
- [40] David Sobrin-Hidalgo, Miguel Ángel González-Santamarta, Ángel Manuel Guerrero-Higuera, Francisco Javier Rodríguez-Lera, and Vicente Matellán-Olivera. 2024. Enhancing Robot Explanation Capabilities through Vision-Language Models: a Preliminary Study by Interpreting Visual Inputs for Improved Human-Robot Interaction. arXiv:2404.09705 [cs.RO] <https://arxiv.org/abs/2404.09705>
- [41] Mohan Sridharan. 2023. Integrated Knowledge-based Reasoning and Data-driven Learning for Explainable Agency in Robotics. In *Explainable Agency in Artificial*

*Intelligence: Research and Practice*, David Aha and Silvia Tulli (Eds.). CRC Press, to appear.

- [42] Alessandro Umbrico, Amedeo Cesta, Gabriella Cortellessa, and Andrea Orlandini. 2020. A Holistic Approach to Behavior Adaptation for Socially Assistive Robots. *International Journal of Social Robotics* 12, 3 (01 Jul 2020), 617–637.
- [43] Luyao Yuan, Xiaofeng Gao, Zilong Zheng, Mark Edmonds, Ying Nian Wu, Federico Rossano, Hongjing Lu, Yixin Zhu, and Song-Chun Zhu. 2022. In situ bidirectional human-robot value alignment. *Science Robotics* 7, 68 (2022), eabm4183.
- [44] Ceng Zhang, Junxin Chen, Jiatong Li, Yanhong Peng, and Zebing Mao. 2023. Large language models for human–robot interaction: A review. *Biomimetic Intelligence and Robotics* 3, 4 (2023), 100131.