Explainable Goal-driven Agents and Robots - A Comprehensive Review

FATAI SADO, CHU KIONG LOO, and WEI SHIUNG LIEW, University of Malaya, Malaysia
MATTHIAS KERZEL and STEFAN WERMTER, University of Hamburg, Germany

Recent applications of autonomous agents and robots have brought attention to crucial trust-related challenges associated with the current generation of **artificial intelligence (AI)** systems. AI systems based on the connectionist deep learning neural network approach lack capabilities of explaining their decisions and actions to others, despite their great successes. Without symbolic interpretation capabilities, they are 'black boxes', which renders their choices or actions opaque, making it difficult to trust them in safety-critical applications. The recent stance on the explainability of AI systems has witnessed several approaches to **eXplainable Artificial Intelligence (XAI)**; however, most of the studies have focused on data-driven XAI systems applied in computational sciences. Studies addressing the increasingly pervasive goal-driven agents and robots are sparse at this point in time. This paper reviews approaches on explainable goal-driven intelligent agents and robots, focusing on techniques for explaining and communicating agents' perceptual functions (e.g., senses, vision) and cognitive reasoning (e.g., beliefs, desires, intentions, plans, and goals) with humans in the loop. The review highlights key strategies that emphasize transparency, understandability, and continual learning for explainability. Finally, the paper presents requirements for explainability and suggests a road map for the possible realization of effective goal-driven explainable agents and robots.

CCS Concepts: • **Computing methodologies** → **Intelligent agents**; *Knowledge representation and reasoning;*

Additional Key Words and Phrases: Accountability, continual learning, deep neural network, explainability, explainable AI, goal-driven agents, transparency

211

1 INTRODUCTION

1.1 Background/Motivation

One of the goals of human-robot partnership is to establish teams of humans and robots that effectively make use of the individual capabilities of each team member [184]. Robots are used in situations involving boring and repetitive work or for tasks that involve a significant risk of injury. Humans, on the other hand, play the role of supervisor, operator, mechanic, bystander, or teammate [184]. Having more than one robot member being supervised by one human should improve the efficiency and productivity of the team. Hence, giving the robots more autonomy and the ability to act semi-independently would give the human more freedom to supervise multiple robots simultaneously.

Goal-driven artificial intelligences (GDAIs) include agents and robots that are autonomous, capable of interacting independently within their environment to accomplish some given or self-generated goals [17]. These agents should possess human-like learning capabilities such as perception (e.g., sensory input, user input) and cognition (e.g., learning, planning, beliefs). GDAIs engage in tasks that require activity over time, generate plans or goals, and execute their ideas in the environment, applying both perception and cognition. They can also adapt the intentions or purposes as the need arises and may be required to account for their actions [31, 124] and embody such capabilities in the context of lifelong developmental learning [218]. GDAIs are helpful in several applications including exploration in hazardous areas, transportation, and gaming [31].

In addition to problem-solving skills, the agents should provide a causal explanation for their decisions and reasoning. A relevant example in this context involves an autonomous robot that plans and carries out an exploration task. It then participates in a debriefing session to provide a summary report and addresses a human supervisor's questions. Agents must explain the decisions they made during plan generation, stating considered alternatives, to report which actions they executed and why, explain how actual events diverged from the plan and how they adapted in response, and communicate decisions and reasons in a human-understandable way to gain the user's trust [14, 41, 200].

This review focuses on two aspects of human-like learning for GDAIs: explainability and continual learning of explanatory knowledge. We focus on both situated/non-situated and embodied/non-embodied autonomous **eXplainable GDAIs (XGDAIs)**. "Situated agents" are located within the environment that they can perceive and operate in, as opposed to "non-situated agents" which are located in a remote location. "Embodied agents" have a physical form while "non-embodied agents" are purely digital.

The review categorizes explanation generation techniques for XGDAIs according to the agent's behavioral attributes, i.e., reactive, deliberative, and hybrid. It provides a clear taxonomy of XG-DAIs and clarifies the notion of the agent's behavioral attributes that influence explainability. For each category, the review focuses on explanation at the level of the agent's perception (e.g., sensory skills, vision) and cognition (e.g., plans, goals, beliefs, desires, and intentions). While an agent's perceptual foundation (dominated by reactive reasoning) may be connected to the sub-symbolic reasoning part relating the agent's states, vision, or sensors/environmental information to the agent's cognitive base, the cognitive base (dominated by deliberative reasoning) relates plans, goals, beliefs, or desires to executed actions. Finally, we provide a road-map recommendation for the effective actualization of XGDAIs with an extended perceptual and cognitive explanation capability.

1.2 What Is eXplainable AI And Why Is Explainability Needed?

XAI refers to machine-learning or artificial intelligence systems capable of explaining their behavior in human-understandable ways [59]. Explanations help humans collaborate with an

autonomous or semi-autonomous agent to understand why the agent fails to achieve a goal or unexpectedly completes a task. For instance, if an autonomous agent fails to finish a task or performs an unexpected action, the human collaborator would naturally be curious to understand how and why it happens. Explanations thus enable the human collaborator to comprehend the dynamics leading to the agent's actions, enabling the human to decide how to deal with that behavior.

Although the need for explainability in AI systems has been long established during the MYCIN era, also known as the era of the expert systems [167, 168], the current drive for explainability has been motivated by recent governmental efforts from the European Union, United States (USA) [29], and China [150], which have identified AI and robotics as economic priorities. The European Union's key recommendation is the right to explanation which is stressed by the **General Data Protection Regulation (GDPR)** [8, 31]. AI systems must explain their decisions, actions, or predictions for safety-critical applications to ensure transparency, explainability, and accountability [150].

1.3 What Is Data-driven And Goal-driven XAI?

In machine learning, explainability in data-driven AI is often linked to interpretability. According to Biran and Cotton [17], a particular system is interpretable if a human can understand its operations through explanation or introspection. Choo and Liu [41] describe a **deep learning neural network's (DNN)** interpretability as determining the input attributes that account for output predictions. Thus, data-driven XAI implies explaining the decision made by a machine learning "blackbox" system, given input data [81]. The motivation to discover how available data contributes to a decision is an important aspect of this branch of XAI and whether the machine learning process can reliably replicate the same decision, given similar data and specific circumstances [95].

On the other hand, goal-driven XAI is a research domain that aims to create explainable robots or agents that can justify their behaviors to a lay user [8]. The explanations would assist the user in creating a **Theory of Mind (ToM)**, comprehending the agent's behavior, and contribute to greater collaboration between the user and the agent. A better ToM of the agent would enable the user to understand the agent's limitations and capabilities, thus enhancing confidence and safety levels and preventing failures. The absence of adequate mental models and understanding of the agent can contribute to a failure of interactions [8, 35]. Hase and Bansal [86] study how techniques used for generating explanations affect the users' ability to predict what the explaining agent will do in a given situation. In addition, they discover that what a user considers a good explanation is not the same as how helpful the explanations are. As discussed in the next subsection, there is a growing application of goal-driven XAI in the current AI-dependent world [125].

1.4 Emerging Trends In XGDAI

Figure 1 shows the distribution of works on XGDAIs from 2016 to 2020 by selected application domains. Literature search for relevant works was performed using the search engines Google Scholar and Science Direct. Combinations of search keywords were used using Boolean operators and quote marks to locate the relevant terminology. For example, ("artificial intelligence") AND ("explainable" OR "transparent" OR "legibility" OR "interpretability") AND ("agent" OR "robot") AND ("goal-driven"). The search results were further filtered by examining the text of the abstracts to fulfil the following criteria:

(1) The paper was published between 2016 to 2020 and the full text must be available online.
(2) The paper must be relevant to the explainable AI domain.
(3) The paper must be a primary study, i.e. introducing a novel model, method, or architecture. Survey and review papers were excluded.
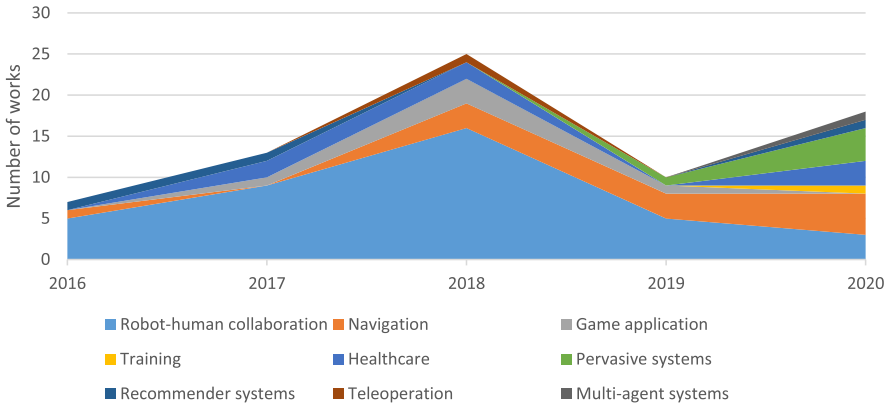
Fig. 1. Five-year trend of the use of explainable techniques in selected application domains.

(4) The paper must involve the use of robots, agents, or AI systems to fulfil a specific function, i.e. "goal-driven". The terminology is not a requirement, however. Several works that fulfil this criterion but do not explicitly call it "goal-driven" were included.

(5) The paper must include explainability methods that are explicitly communicated to the user. Methods that fulfil this criterion include natural language explanations, visualizations, and so on.

In 2016, research into XGDAIs focused on human-robot collaborative tasks, navigation, and healthcare. However, studies in XGDAIs branched out to other domains over the next few years. This upsurge can be seen as the effect of the public pressure on the explainability of AI systems and initiatives by several national government agencies like the "right to explanation" by the GDPR [8, 31]. This trend is likely to continue as AI adoption progresses in other fields. Research into explainability for pervasive systems, for example, can be attributed to the growth of increasingly "smart" and complex systems for both domestic and industrial use. Research into navigation systems is also growing alongside interest in self-driving vehicles. The studies are shown in Table 1 in the Supplementary Materials along with significant taxonomies.

2 TERMINOLOGY CLASSIFICATION

2.1 Attributes Of XGDAI

The studies obtained from the literature search as described in the previous section were taxonomized based on several categories:

(1) Behavioral architecture: whether the architecture of the model was deliberative, reactive, or hybrid.
(2) Application domain: what the model was used for.
(3) Transparency: whether the internal workings of the model were observable or explainable.
(4) Domain specificity: whether the model can be used for only one domain or was domain-agnostic.

Regarding the agent's behavior and interaction with the world, three behavioral architectures are traditionally distinguished in literature for XGDAI: deliberative - in which the agent deliberates on its goals, plans, or actions, or acts based on a sense-model-plan-act cycle (the agent, in this case, should possess a symbolic representation of the world); reactive - in which the agent implements some simple behavioral patterns and reacts to activities or events in its environment in a

stimulus-response way, no model of the world is required (the robot chooses one action at a time), and hybrid - which combines the above two behaviors [53]. Each of the three behavioral architectures can be further sub-divided based on the explainability method used, such as goal-driven autonomy, goal-driven agency, **Belief-Desire-Intention (BDI)**, and so on. Table 1 in the Supplementary Materials presents the taxonomies of XGDAI behavior found in the literature. In this section, we make further clarification of these attributes.

Deliberative. Deliberative agents act more like they think, searching through a behavioral space, keeping an internal state, and predicting the consequences of an action. Wooldridge [221] describes such an agent as one with a symbolic model of its world that could make decisions based on symbolic reasoning or rationale. According to the conventional approach, the agents' cognitive aspect consists of two parts: a planner and a world model [53]. The world model is an internal representation of the agent itself in the external environment. The planner model uses this representation to make a strategy for how the agent's goals can be accomplished. How such agents operate may be viewed as a 'sense-model-plan-act' behavior. The BDI model is the most widely used architecture to enforce such actions, where the beliefs of an agent about the world (its picture of the world), desires (goal), and intentions are internally expressed, and realistic reasoning is applied to select an action [90].

Reactive. Reactive agents present a collection of simple behavioral patterns that react to environmental changes [53], no model of the world is included. They achieve their goal by reacting reflexively to external stimuli, choosing one action at a time. The creation of purely reactive agents came at the heels of the limitations of symbolic AI. Developers of reactive agent architectures rejected symbolic representations and manipulation as a base of AI [113]. Model-free (deep) **reinforcement learning (RL)** is a state-of-the-art approach that enables reactive agent behavior. Some notable explainability works in RL include **Memory-based eXplainable Reinforcement Learning (MXRL)** [47], **Minimal Sufficient eXplanation (MSX)** via Reward Decomposition [103], and **Reward Augmentation and Repair through Explanation (RARE)** [201]. Previous reviews on reactive RL agents can be found in Lin [135].

Hybrid. A significant number of researches have focused on combining reactive and deliberative agent techniques, leading to creating a compound called a hybrid agent, which combines comprehensive internal manipulation with non-trivial symbolic constructs and external events with reflexive reactive responses [53]. The modern drive integrates reactivity's flexibility and robustness with deliberation's foresight [118]. Supporters of the hybrid approach believe it is optimal since high-level deliberative reasoning and low-level response to perceptual stimuli seem essential for actualizing an agent's behavior. An example is the hybrid system proposed by Wang et al. [213]. Reactive exploration is used to create waypoints, which are then used by a deliberative method to plan future movements in the same area. Another existing system which mixes reactive and deliberative behaviors is the agent developed by Rao et al. [172], which determines when to be reactive and when to pursue goal-driven plans.

2.2 Application Scenarios For XGDAI

This section presents a number of application scenarios for XGDAI that are primarily reported in the literature. These are summarized in Table 1 in the Supplementary Materials.

- **Robot-human collaborative tasks** involve close interactions with humans in factory settings, and teaming in an outdoor setting are the predominantly mentioned application scenarios in literature. In robot-human collaborative scenarios, explainability (i.e., transparency) of XGDAI was shown to enhance the quality of task performance [26] and to enable

both robots and humans to take responsibility (credit or blame) for their actions in collaborative situations [111].

- **Robot navigation**, where the robot or agent-controlled vehicle moves through an environment towards a goal while avoiding obstacles. Examples of navigation applications include solving a maze or self-driving vehicles. Explanations are generated in this domain to explain why specific routes are chosen over others or what obstacles are detected and avoided.

- **Game applications** use explainability techniques to describe gameplay behavior, whether of **non-player characters (NPCs)** [151] or player characters [142]. The latter is used to decompose the behavior of human-controlled players into discrete elements that can then be programmed into computer-controlled agents to emulate human gameplay.

- **Healthcare** applications employ agents and robots for medical diagnosis and as caregivers. Autonomous agents in this field are strictly regulated as any error may potentially harm a patient. Using explainer systems, the agents' steps to arrive at a diagnosis can be traced. Medical experts acting as "human-in-the-loop" can observe the agents' justifications and correct them if an erroneous diagnosis is performed.

- **Training** or teaching scenarios involve agents in the role of either instructors or students. As instructors, the agents analyze students' strengths and weaknesses to create individualized teaching syllabi. As students, agents emulate human behavior of, for example, pilots in cockpits. Explanation generation methods are used for the latter to diagnose their reasoning and decisions.

- **Pervasive systems** or ubiquitous computing utilize agents to manage highly complex systems and monitor many variables simultaneously. Explainability simplifies the system operations, making it easier and quicker for human operators to intervene and diagnose errors [210].

- **Recommender systems** are typically user interfaces to match a user's preferences to a large database of products. Given a large number of user variables, the system searches for the closest products that fulfill all user requirements. Recommenders with explainability systems can provide a larger selection of products and explain how each product matches the user's preferences.

- **Multi-agent systems** relate to using multiple agents cooperating to accomplish a goal. Emergent behavior occurs when interactions of multiple agents produce outcomes unforeseen by the programmers. Explainer systems provide insights on how and why individual agents take certain actions and how the cumulative choices produce emergent behavior [92].

2.3 Transparency

Transparency refers to whether the internal workings of the XGDAI are readily observable. Observable processes should include from the moment the model receives sensory input to when the model produces the final output decision, as well as every intermediate step. In addition, the observable elements should be in a format that is easily interpretable, for example using natural language to describe the processes or visualization techniques to highlight significant areas. Transparent models by their nature should be replicable; a human user should be able to explain how and why a specific input arrives at a specific output.

3 EXPLANATION GENERATION TECHNIQUES FOR XGDAI

This section presents existing explanation generation techniques and taxonomies (e.g., transparency, domain dependence, post-hoc explainability, continual learning) for XGDAI. The section is subdivided into three parts to discuss techniques categorized under Deliberative, Reactive, or Hybrid Explanation Generation. The techniques can be further categorized by Transparency
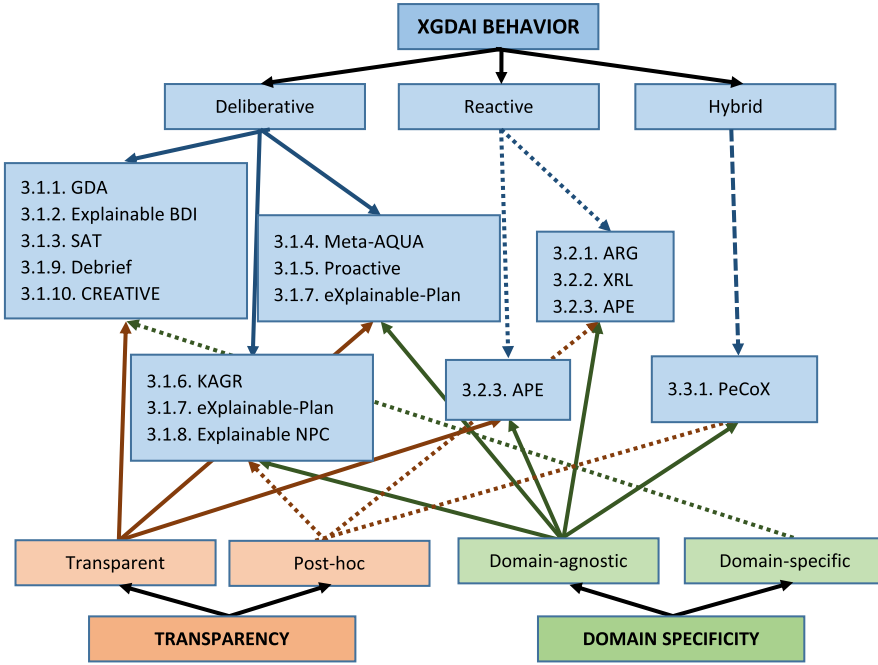
Fig. 2. Taxonomy of the reviewed XGDAI.

(Transparent vs Post-hoc) and Domain Specificity (Specific vs Agnostic). Domain-specific techniques heavily depend on the agent world's domain knowledge and do not permit application to other agents in other environments. Domain-agnostic techniques are domain-independent, allowing cross-platform usability. Post-hoc explanations make it possible to explain without inherently following the reasoning process leading to the decision [8]. Transparent explanations, on the other hand, highlight every step of the reasoning process. Figure 2 maps the reviewed XGDAI to three taxonomy categories: XGDAI Behavior, Transparency, and Domain Specificity.

3.1 Deliberative XGDAI

From the literature search, models are classified as Deliberative if they have two components: an internal representation of the external environment, and a planner to design a strategy to accomplish the agent's goals with respect to the environment.

3.1.1 Models of Goal-driven Autonomy - Transparent Domain-agnostic. Goal-driven autonomy is a conceptual goal justification mechanism that enables an agent to generate explanations when detecting a mismatch in the environment state. The model enables the robot to continuously track the execution of its current plan, determine whether the states met expectations, and generate an explanation upon identifying a mismatch [152]. The model is also extended to enable an agent to identify when new goals should be chosen and explain why new goals should be followed [99].

In many existing works in goal-driven autonomy, explanations are generated when active goals are completed or when discrepancies are detected between the actual event and the intended plan. Differences may arise due to one of several factors; if the domain knowledge is flawed, if the perception of a state is inaccurate, or if there is an unseen factor influencing the state. An explanation is generated to explain the discrepancies and find or address the hidden factors [152].
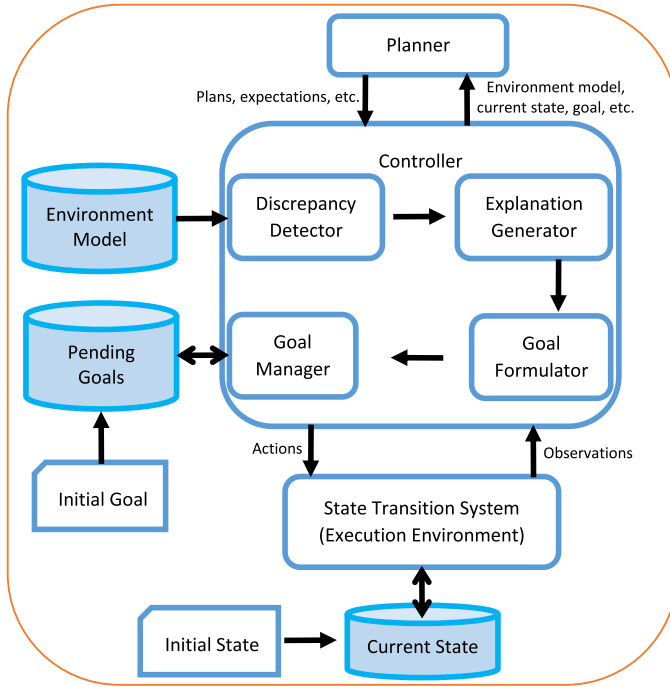
Fig. 3. The ARTUE goal-driven autonomy conceptual model [152]. The model extends the Nau [156] online planning model. The controller communicates with the execution environment (i.e., state transition system) and the planner. As input, the planner receives a planning problem consisting of an environmental model, the current state of the system, and a potential goal. The planner generates a series of action plans and expectations (states constraints when executing actions). The controller sends the action plan to the state transition system, then processes the subsequent observations. The controller detects discrepancies, produces explanations, generates goals, and manages the goals.

The **Autonomous Response To Unexpected Events (ARTUE)** [152] is a domain-independent goal-driven agent that continuously reacts to unexpected situations by reasoning on what goals to pursue to resolve the situation. ARTUE is designed to deal with unexpected environmental situations by first explaining these changes and then developing new goals which integrate the explicit information on the hidden environmental aspects. On this basis, ARTUE can deal with new and unobserved objects in the planning. The ARTUE system is implemented in a sandbox simulation scenario [11]. The ARTUE system as shown in Figure 3 integrates a planner that deliberates on exogenous events through predicting future states in a dynamic environment, and a controller that identifies discrepancies, generates explanations, generates targets, and manages goals. The explanation aspect deliberates on the world's hidden knowledge by abductive reasoning on the conditions and effects of the planning [174].

3.1.2 Explainable BDI Model - Transparent Domain-specific. BDI agents or robots, primarily symbolic AIs with integrated beliefs, desires, and intentions, offer good prospects for producing useful and human-understandable explanations [85]. According to Bratman [25], belief and desire are mental attitudes (pro-attitudes) driving action. Intention is distinguished as conduct controlling this pro-attitude, which can be treated as elements of partial plans of action [25, 72]. Harbers et al. [85] propose a BDI model that allows the explanation of a BDI agent's actions (behavior) based on the underlying beliefs and goals. Goals are described as "active" desires that the agent

currently pursues. An example of an explanation of actions based on 'belief' can be: "The boy opened the window because he believed someone was outside", and based on 'goals' can be: "The boy opened the window because he wanted to see who was outside". The motivation here is that humans can clarify and comprehend their behaviors or actions in terms of underlying beliefs, desires, intentions, or goals [107, 143]. Thus, since BDI agents establish their actions by deliberation on their mental state, the mental principles behind the action are applied to interpret their actions. Also, since mental reasoning is expressed symbolically, the explanation generation process is straightforward. Typically, a log of behavior preserves all the agent's prior mental states and actions that could be used for explanations. On request, the BDI algorithm is implemented on the log and selects the beliefs and goals that become part of the explanation. However, not all 'explanatory elements' can be helpful in the explanation [107].

An important aspect of explainable BDI agents is that they can clarify typical human errors. According to Flin and Arbuthnot [65], revealing explainable BDI agents' actual mental state to trainees may make them aware of their (false) assumptions about them. People may make false assumptions about others' experiences and motives in many crucial circumstances [65], a phenomenon of attributing incorrect mental states to others [110].

3.1.3 **Situation Awareness-based Agent Transparency (SAT)** *Model - Transparent domain-specific.* SAT is an agent transparency model for enhancing an operator's situational awareness [60] in the agent's environment [23, 39]. The SAT model depends on the BDI model framework and is implemented as a user interface to provide transparency to the operator. An important extension is that it provides transparency not only of the status of the robot (e.g., plans, current state, and goals) and process of reasoning but also of the future projections of the robot (e.g., future environment states) [38]. Basic information on the robot's current goal and state, intentions, and expected action is presented to the operator at the entry phase of the SAT structure. In the second stage, knowledge about the agent's reasoning process supporting its action and the limitations it takes into consideration are presented before the operator. In the third stage, specifics of the robot's future projection are given to the operator, such as expected outcomes, the possibility of failure or success, and any ambiguity inherent in the predictions. The transparency of an agent in this context is an informative ability to provide an operator's understanding of the purpose, results, plans, reasoning process, and future projections of an agent. The operator's trust is described as the agent's willingness to help achieve the operator's goals in especially uncertain and vulnerable circumstances [8]. SAT requires highly specific information with regards to the agent, task, and environment combined to create a dynamic construct. As each new input arrives, the SAT incorporates both temporal ("what happened when") and spatial information ("what happened where"). The internal model integrates various information and understands their context relevant to goals in order to come up with appropriate actions for the situation. Due to the significant effort required for designing the SAT model to fit the application, the method was considered domain-specific.

3.1.4 *Meta-AQUA Explanation Model - Transparent Domain-agnostic.* Meta-AQUA is an introspective learning system proposed by Cox [45] for a self-aware cognitive agent. The system can allow an agent to decide (learn) its goals by understanding and describing unusual occurrences in the world. The learned objectives seek to modify the agents' perception of the world to minimize the dissonance between what the agent expects and the world's actual reality. With intelligent behaviors, Meta-AQUA integrates cognitive components such as planning, understanding, and learning and meta-cognition such as cognition monitoring and control. Learning in this context is described as a deliberate planning task with learning goals.

In contrast, the explanation of unusual events is a key to enhancing the agent's learning of a new goal. Meta-AQUA uses its meta-reasoning component (i.e., meta-cognition) to explain a reasoning
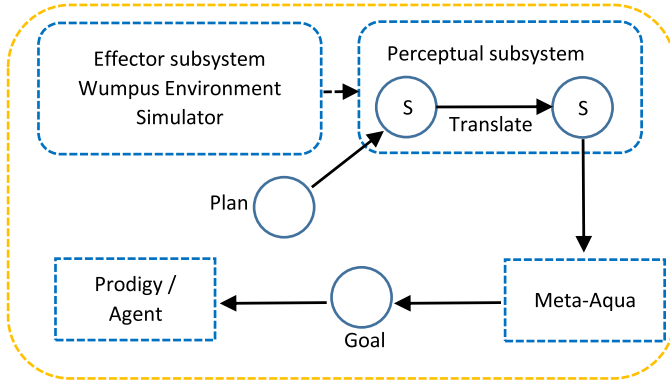
Fig. 4. The INTRO architecture [45] consists of a primitive subsystem of perceptions and effectors and two components of cognitive planning and comprehension.

or expectation failure, enhancing its goal learning. It adopts case-based knowledge representations as frames linked together by explanatory patterns to represent general causal structures [45, 46].

Meta-AQUA is implemented in the **INitial inTROspective cognitive agent (INTRO)** and simulated in the Wumpus World-simulated environment, a partially observable environment where the objective of the agent or robot is to seek a pot of gold while evading pits and the Wumpus creature (Figure 4). INTRO is designed with primitive perceptual and effector subsystems and two cognitive components.

INTRO has four main components: a primitive subsystem of perceptions and effectors and two components of cognitive planning and comprehension. The cognitive components are composed of the respective prodigy/agent and Meta-AQUA structures. The perceptual subsystem acts as a translator for the cognitive subsystems. The central cognitive cycle is to explore the world, form an objective of changing the world by understanding and describing unusual events or world states, establish a plan to accomplish the goal, and finally perform according to the plan, observing the outcomes. Instead of using explanation to alter its understanding, INTRO uses explanation to create a goal to improve the environment.

The agent will observe the world with these components, develop a goal to transform the world, devise a strategy for achieving the objective, and eventually change the environment, including turning, going forward, picking up, and firing an arrow. As the agent navigates the environment, the Meta-AQUA system seeks to explain the effects of the agent's behavior by inserting the events into a conceptual representation and constructs an internal model to reflect the causal ties between them. Meta-AQUA creates an anomaly or other interesting occurrence to produce an explanation of the event. However, a major aspect of this approach is that the agent utilizes the explanation to produce a goal to modify the environment (e.g., shoot an arrow) instead of using the explanation to change its understanding.

3.1.5 *Proactive Explanation Model - Transparent Domain-agnostic.* In scenarios that involve teams of humans and autonomous agents, a proactive explanation that anticipates potential surprises can be useful. By providing timely explanations that prevent surprise, autonomous agents could avoid perceived faulty behavior and other trust-related issues to collaborate effectively with team members. This is an important extension to explainability for agents since most techniques are usually provided to respond to users' queries. In this context, Gervasio et al. [73] present a conceptual framework that proposes explanations to avert surprise, given a potential expectation violation. The proactive explanation attempts to reconcile the human mental model with the agent's
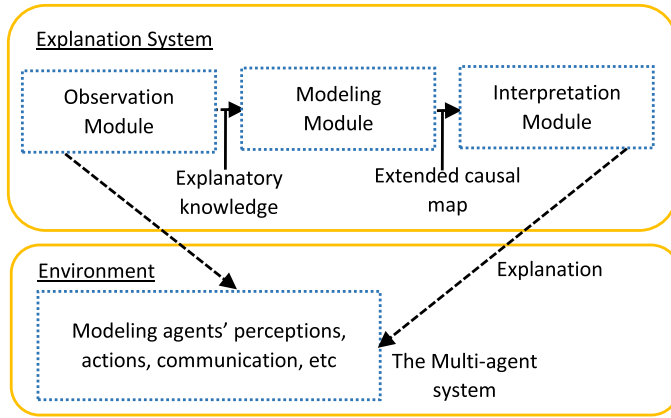
Fig. 5. KAGR explanation architecture [183]. The system consists of an observation module for the generation of the explanatory knowledge, a modeling module for the knowledge formalism using an extended **causal map (CM)**, and an interpretation module to produce explanations.

real decision-making process to minimize surprise that could disrupt the team's collaboration. Surprise is the primary motivation for pro-activity and is triggered if the agent's action deviates from its past actions, or if its action is considered unusual or contrary to plan [73].

3.1.6 KAGR Explanation Model - Post-hoc Domain-agnostic. KAGR is an explanation structure proposed by Sbai et al. [183] to explain agent reasoning in **multi-agent systems (MAS)** (operating in uncontrollable, dynamic, or complex situations) where an agent's reasoning is not reproducible for the users. KAGR describes an agent's reasoning state at run-time as a quadruplet **<K,A,G,R> (Knowledge, Action, Goal, Relation)**, also considered explanatory knowledge. Under the KAGR conceptual explanation framework, events produced by the agents at run-time are intercepted, and an explanatory knowledge acquisition phase is performed to represent the knowledge attributes or details related to the execution of events in a KAGR structure [91, 183]. A further step is performed to generate a knowledge representation formalism by linking the attributes in an extended CM model. A final step achieves natural language interpretation for the CM model using predicate first-order logic to build up a knowledge-based system for an understandable reasoning explanation to users. Overall, as seen in Figure 5, it adopts a three-module architecture: a first module that generates explanation, another module that puts the knowledge in the formalism of the extended causal chart, and a third module that uses first-order logic to analyze and interpret the constructed CM to produce reasonable explanations.

3.1.7 eXplainable Plan Models - Post-hoc/Transparent Domain-agnostic. An aspect of planning is Plan Explanation, in which the primary objective is to enable individuals to understand the plans of the (agent) planner [16, 185, 193], including interpreting the agents' plans into human-understandable form and the creation of interfaces that facilitate this understanding. In this context, related works include XAI-Plan [21], **Refinement-Based Planning (RBP)** [16], and WHY-Plan [118].

XAI-Plan is an explanatory plan model proposed by Borgo et al. [21] to provide an immediate explanation for the decisions taken by the agent planner. The model produces explanations by encouraging users to try different alternatives in the plans and compares the subsequent plans with those of the planner. The interactions between the planner and the user enhance hybrid-initiative planning that can improve the final plan. The XAI-Plan planning system is domain-agnostic and
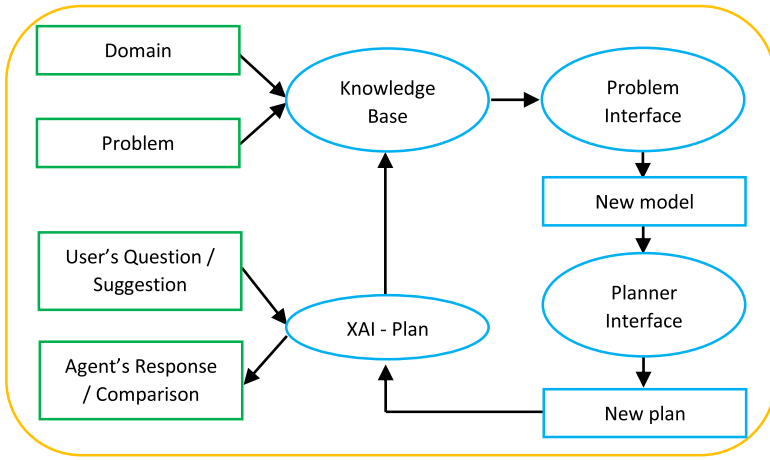
Fig. 6. Example XAI-Plan architecture by Borgo et al. [21]. The XAI-Plan node generates explanatory plans and communicates the agent's response through a user interface. ROSPlan provides the planner interface, problem interface, and knowledge base.

independent. XAI-Plan answers questions such as "why does the plan include a specific action and not another similar action?". The algorithm uses a preliminary set of plans as input; the user chooses one of the plan's actions; the XAI-Plan node generates explanatory plans and communicates with the user through the user interface (Figure 6). ROSPlan provides the planner interface, problem interface, and knowledge base, which is used to store a **Planning Domain Definition Language (PDDL)** model and provide the AI planner (i.e., an architecture for embedding task planning into ROS systems) with an interface.

RBP is a transparent domain-independent framework proposed by Bidot et al. [16] to allow verbal queries from users and produce plan explanations verbally. RBP enables a transparent description of the search space examined during the planning process, giving the possibility to explore the search space backward to search for the relevant flaws and plan modifications. RBP builds on a hybrid planning structure incorporating hierarchical planning and planning using partial-order causal-link using states and action primitives [18].

Korpan and Epstein [118] propose WHY-Plan as an explanation method that contrasts the viewpoints of an autonomous robot and a human while planning a navigation route. The core of its explanation is how the planners' objectives differ. WHY-Plan addresses the question "Why does your plan go this way?" and exploits differences between planning objectives to produce meaningful, human-friendly explanations in natural language. An example of a WHY-Plan natural language explanation for a robot, in a team with a person, navigating from one location to another and avoiding a crowded path could be, "Although a somewhat shorter route exists, I think mine is a lot less crowded". The WHY-Plan's response compares two objectives: "avoid crowds", its planning objective, and an alternative objective: "take the shortest path", which can be attributed to the human questioner or team member. WHY-Plan is implemented in SemaFORR, a cognitive-based hybrid robot controller [61]. SemaFORR can make a decision using two sets of reactive advisors.

3.1.8 Explainable **Non-Player Characters (NPC)** *- Post-hoc Domain-agnostic.* Explainable NPC is an architecture proposed by Molineaux et al. [151] to minimize the frustration of video game players by explaining NPC actions. To many video game players, NPCs may become a source of frustration because their reasoning process is not transparent. The NPCs may respond to some internal necessity that a player is unaware of or may face barriers that a player cannot see, but
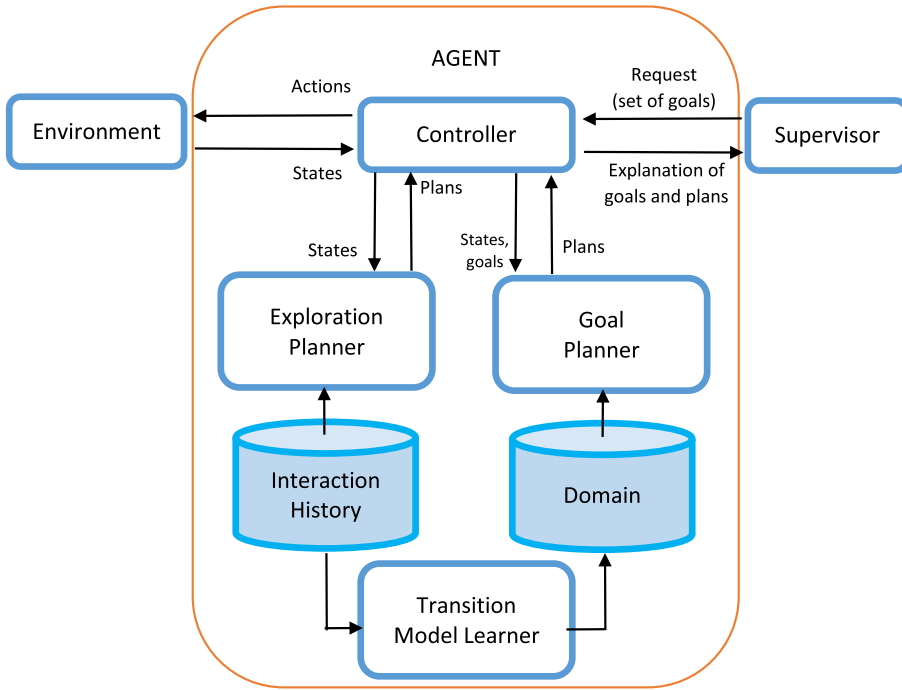
Fig. 7. Example of an Explainable NPC architecture [151]. The agent has four sub-modeled functionalities: the exploratory planner for obtaining new information; the goal planner necessary for implementing the supervisor's goals; the transition model learner responsible for updating the world model of the agent; and the controller responsible for determining when to explore.

they do not communicate these problems [151]. The NPC architecture is thus motivated to allow agents to know their environment, achieve goals and demonstrate to a supervisor what they are doing (Figure 7). At each step, the architecture enables the agent to obtain an observation that represents knowledge about the environment's true state and to communicate with the environment (and supervisor) by taking action. The supervisor requests the agent that represents what the supervisor wants the agent to achieve, modified at each step. In response, the agent should always clarify why it takes a specific action to the supervisor. The framework is divided into four sub-modules: the exploratory planner, tasked with taking steps to collect new data to update an action model; the goal-directed planner, necessary for implementing the supervisor's goals; the learner of the transition model, necessary for updating the world model of the agent; and the controller, responsible for deciding when to explore [151].

In related work, Van Lent et al. [208] propose the XAI architecture for NPCs in a training system. The XAI functions to retrieve important events and decisions made from the replay log and enable NPCs to explain their conduct as an answer to the questions selected from the XAI menu. In this framework, each character's reactive behavior, low-level actions, and higher-level behavior generation are the responsibility of the NPC AI. The XAI system records the NPC AI's activities and utilizes the records or logs for post-action analyses during the execution process.

3.1.9 Debrief - Transparent Domain-specific. Debrief is a multimedia explanation system proposed by Johnson [102] that constructs explanations after an active goal. The circumstance in which the decision was taken was recalled and then replayed in the variants of the original

situation, or by experimenting to determine which factors influenced the decision. The elements are considered critical as their absence would result in a different outcome of the decision process. The details of the agent's implementation, including individual rules and motivations applied in making a decision, are filtered out to construct an explanation. Under the framework, it is unnecessary to maintain a complete trace of rule firings to produce explanations. The system learns the relationships between situational factors and decisions to facilitate the explanation of similar decisions. Debrief is implemented in an artificial pilot simulator where the pilot can perform a flight patrol mission. After each task, the pilot is asked to describe and explain essential decisions it made along the way. The pilot has to clarify its evaluation of the situation. Debrief should enable the pilot to define the reasons for its actions and assess the situation and its beliefs that resulted in the actions.

3.1.10 Explainable CREATIVE Model - Transparent Domain-specific. **Cognitive Robot Equipped with Autonomous Tool InVention Expertise (CREATIVE)** is a relational approach proposed by Wicaksono and Sheh [219], enabling robots to learn how to use tools and design and construct new tools if needed. The critical information, such as learned observations, snapshots from a camera, and object positions, are stored in **Prolog (Programming in Logic)** to simplify the explanation of its tool creation process. CREATIVE utilizes relational representation, so its results have some inherent explainability features, establishing the relation between entities as facts in Prolog. A method of **inductive logic programming (ILP)** is employed to learn the relational representations of the tool models [127]. A pre-determined set of basic questions and answers is used in the entire dialogue in Prolog.

3.1.11 Summary on Deliberative XGDAIs. Overall, reviewed literature on deliberative XGDAIs provides applicable explanation generation models and frameworks focusing on transparency and/or comprehensibility of agents' plans, goals, actions (behavior), reasoning processes, decisions, current states, and world states (i.e., unusual events and future projections of states). Leveraging on their model for plan execution or model of the world or tasks, their primary triggers or key basis for explanation generation include factors such as expectation failures (i.e., mismatches between what the agent expects and what the agent encounters), unusual events (in the world that is not previously modeled for the agent), anticipated surprise for the agent's action or decision, mismatched mental models (of the agent and human), users' requests, and an underlying belief or goal driving an action. These techniques assume certain aspects of the human-like explanation generation process. However, most approaches are domain-dependent, over-fitted to the particular situation or environment, limiting application to other domains.

Another concern for deliberative XGDAIs is how explanatory knowledge is acquired for explanation generation. Handcrafting explanatory knowledge or storing logs of agents' internal states (e.g., mental states and actions) or Prolog of facts appears to be the predominant technique. In this context, resource management for storing explanatory knowledge is a significant concern that is yet to be addressed in the literature. A summary of key findings on deliberative XGDAIs is presented in Table 2 in the Supplementary Materials.

3.2 Reactive XGDAIs

From the literature search, models were classified as Reactive for having a set of simple behavioral patterns for reacting to environmental stimuli, without needing to maintain an internal world model of the environment.

3.2.1 ***Automated Rationale Generation (ARG)*** *Model - Post-hoc Domain-agnostic.* ARG is an explainable model proposed by Ehsan et al. [59] for real-time explanation generation, which
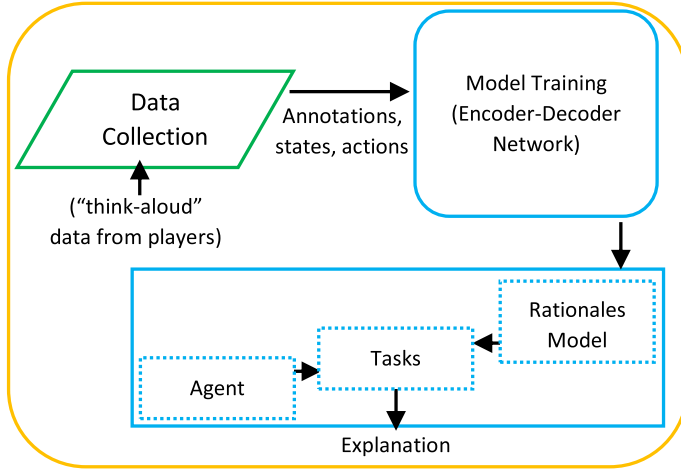
Fig. 8. The ARG architecture [59] gathers a think-aloud data corpus (i.e., actions, game states, and annotations) of players discussing their actions during the game. Subsequently, the encoder-decoder network is trained for agents to produce rationales for their actions.

is trained to interpret an agent's inner state and data representation in natural language. The approach is domain-agnostic and ideally developed for a 'Markovian' sequential environment where the agent's states and actions are accessible to the model and where decisions (i.e., selection of actions that maximize expected future rewards or utility) made by the agent in the past have an impact on future decisions. ARGs are expected to generate explanations or rationales in the natural language for the agent's behavior as if a person had performed such behavior [58]. The idea behind the rationale generation is that people can communicate effectively by verbalizing reasonable motives for their actions, even though the verbalized reasoning is not aligned with the decision-making neural processes of the human brain [19].

Ehsan et al. [59] apply ARG for human-like explanation generation for an agent playing Frogger, a sequential game environment. The ARG approach is to collect a corpus of human-like explanation data (Figure 8) - i.e., a thought-aloud data corpus of players that discussed their behavior during the game and train a neural rationale generator (an encoder-decoder NN) using the corpus to enable the agents to produce their plausible human-like rationales or explanations.

While the results are promising, the rationale generator's potential limitations may be the lack of a more grounded and technical explanation framework. The framework lacks interactivity that allows users to question a rationale or suggest that an agent clarifies its decision in another way. Another limitation may stem from the data collection process that introduces breakpoints to request an explanation from the players. It is necessary to determine an appropriate method to collect data from players during run-time without interruption of the players or participants.

*3.2.2 **eXplainable Reinforcement Learning (XRL)** - Post-hoc Domain-independent.* XRL is proposed in several studies for a class of reactive RL agents [47], which are model-free agents that determine what to do depending on their current observations [137]. They rely on a simple behavioral policy scheme (i.e., a state-to-action mapping) that allows them to learn a policy based on trial-and-error interaction with their environment [187]. RL agents do not generally reason or plan for their future actions, making it challenging to explain their behavior. An RL agent may eventually learn that one action is preferred over others or that choosing an action is associated with a higher value to attain the goal but would lose the rationale behind such a decision at the end

of the process [187]. Thus, they lack the mechanism that can effectively explain why they choose certain actions given a specific state. Some existing techniques of explainability for RL agents aim to make the decision process during the policy learning process retrievable and explainable. Some examples include MXRL [47], MSX via Reward Decomposition [103], and RARE [201].

MXRL is an XRL strategy proposed by Cruz et al. [47] to enable an RL agent to explain to users why it selected an action over other possible actions. The model uses an episodic memory [187] to save each episode or agent's record of executed state-action combinations, then computes both the likelihood of success (Q-values) as well as the number of transitions within each episode to meet the final target to provide an explanation or reason for selecting an action over the others. MXRL is implemented in two discrete simulated environments: a bounded grid world and an unbounded grid environment with aversive regions. The RL agent can clarify its actions to lay users at any time by using information obtained from memory. MXRL suffers from limitations with regard to the utilization of memory in broad solution spaces.

MSX is an XRL strategy proposed by Juozapaitis et al. [103] to explain RL decision agents via reward decomposition. The idea is to break down incentives into amounts of semantically meaningful reward types, allowing actions to be contrasted in trade-offs between them. MSX should provide a concise description of why one behavior is favored over another in terms of reward styles in a domain-independent setting. It uses an off-policy algorithm known as Q-learning that leads to the best policy and decomposed action values. The focus is on explanations that learn Q-functions that allow observing the action preference of the agent quantitatively. MSX is implemented in two environments to support its validity: a CliffWorld grid-world where cell contents are represented as decomposed reward vectors in the form of [cliff, gold, monster, treasure]; and a Lunar Lander rocket scenario where the actions can be decomposed into natural reward types including crashing penalty, safe landing bonus, main-engine fuel cost, side-engines fuel cost, and shaping reward that defines scenarios close to the actual world of controlling a rocket during a ground landing.

RARE is an extension of the XRL strategy [201] to address the need for establishing a shared behavior mental model between an RL agent and a human collaborator using a timely update to the reward function. The RARE framework is modeled on the premise that sub-optimal collaboration activity results from a misinformed understanding of the task/assignment rather than a problem with the reasoning of the collaborator. Thus, using the Markov decision-making process, sub-optimal human decision-making is attributable to a malformed policy due to an inaccurate task model. RARE will determine the most probable reward function for human actions through interactive learning and reward updates. Subsequently, the missing aspects of the reward function are identified and conveyed as actionable information to allow the collaborator to update the reward function (task understanding) and policy (behavior) while performing the task and not after the task is completed. This mechanism should allow the agent or robot to provide a policy update to a person (i.e., by explaining the correct reward to the human) based on the perceived model difference, minimizing the risk of costly or dangerous errors during everyday tasks. A color-based collaborative version of Sudoku and an autonomous robot are used to implement RARE [201]. The robot is reported to interrupt users who are on the verge of making a mistake, inform them that their actions will cause a failure of the task, and explain which constraint of the game will inevitably be violated. However, the RARE model still lacks the comprehensibility of its optimal policies. During the computation of an optimal policy, the factors taken into account for risk and reward considerations for each state and prospective action are lost.

Other XRL strategies for model-free RL agents include the work of Madumal et al. [142], which utilizes causal models to generate 'contrastive' explanations (e.g., "why" and "why not" questions) as a means of describing partly measurable agent action or actions in a game scenario (Starcraft II).

The approach is to learn a **structural causal model (SCM)** during RL and to generate explanations for "why" and "why not" questions by counterfactual analysis of the learned SCM. However, one weakness of the approach is that the causal model must be given beforehand. In another work by Pocius et al. [166], deep RL is used to provide visual interpretations in saliency maps to describe an agent's decisions in a partially observable game scenario. However, saliency maps have several disadvantages [3]. They do not help to explain long-term causality. The presence or absence of certain critical features may produce significantly different saliency maps. And finally, saliency maps ignore image features that cannot be localized to individual pixels. The study by Sequeira and Gervasio [187] provides explanations through an introspective analysis of the RL agent's interaction history. The framework explores the history of an agent's environmental experiences to retrieve interesting elements that explain its behavior.

3.2.3 **Autonomous Policy Explanation (APE)** *- Post-hoc Domain-agnostic.* APE is a strategy proposed by Hayes and Shah [88] for a class of reactive agent robot controllers that rely on black-box-trained reinforcement learning models [108] or on hard-coded conditional statement-driven policies to enable the robot to autonomously synthesize policy descriptions and respond to natural language queries by human collaborators. The objective is for the robot to reason over and answer questions about its underlying control logic independently of its internal representation, allowing human coworkers to synchronize their perceptions (or mental models) and recognize defective actions in the robot controller. The model applies to discrete, continuous, and complex dynamic environments. Using a **Markov Decision Process (MDP)** model for constructing the domain and policy models of the control software, the system learns a domain model (i.e., collection of states) of its operating environment and the robot's underlying control logic or policy from actual or simulated demonstrations or observations of the controller's execution traces. These are incorporated into a single graphical model that collects the essential details about states' relationships and behavior.

The model employs communicable predicates for natural language communication, i.e., Boolean classifiers similar to traditional STRIPS-style [64] planning predicates with associated natural language descriptions, to translate attributes from sets of states to natural language and back (Figure 9). The algorithms can then enable the agent to answer a direct inquiry about behavior-related questions, e.g., "When do you do _?" - requesting an explanation about the occurrence of a specific controller behavior, or "Why didn't you do _?" - requesting an explanation for why a certain controller activity was not observed.

3.2.4 Summary on Reactive XGDAIs. Overall, reviewed literature on reactive XGDAIs relies on simple behavioral policy models to generate explanation. Most approaches are agnostic and have the potentials to be applied in several domains. However, several gaps are still visible. For example, some models lack a more grounded and technical explanation framework [59] (e.g., an interactive explanation platform, breakpoint-free explanatory knowledge collection process). Some others, particularly RL agents, cannot articulate rationales for their actions or 'concerns' (i.e., comprehensibility of agent's optimal policy after policy convergence). A summary of key findings on reactive XGDAIs can be seen in Table 3 in the Supplementary Materials.

3.3 Hybrid XGDAI

From the literature search, methods are designated as Hybrid XGDAI if they possess elements of Deliberative and Reactive methods. Hybrid agents typically combine the instinctive responses of Reactive methods with the complex reasoning capabilities of Deliberative methods.

3.3.1 **Perceptual-Cognitive eXplanation (PeCoX)** *- Domain-agnostic.* PeCoX is a framework proposed by Neerincx et al. [157] to create explanations that explore the cognitive and
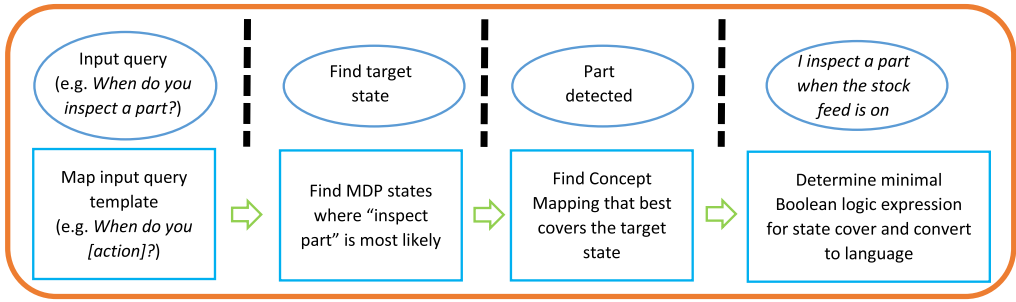
Fig. 9. Automated policy generation framework [88]. The framework maps a policy clarification to an action/input request. An input request or query (e.g., "When do you 'inspect a part'?") is first matched against pre-defined query templates (e.g., "When do you [action]?"). A graphical search algorithm is used to find the state regions that fulfill the query criteria. Using logical combinations of communicable predicates, a concept mapping that best covers the target state is found. The cover is then reduced and translated via the template into language.
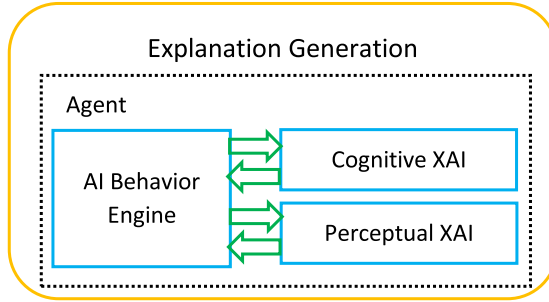

Fig. 10. PeCoX generation framework [157]. The framework distinguishes between Perceptual XAI and Cognitive XAI. The Perceptual XAI is connected to the sub-symbolic reasoning of the AI behavior engine and is designed to clarify the perceptual aspect of the agent's behavior. The Cognitive XAI interacts with this part to ground its belief base. It can explain why certain actions have been selected by linking them to goals or beliefs.

perceptual dimensions of the behavior of an agent (Figure 10). PeCoX's perceptual XAI explains the perceptual aspect of the agent's behavior using a proposed **Intuitive Confidence Measure (ICM)** [207] and a method of contrastive explanations [176] that involve the classification of facts and foils, e.g., 'How did you arrive at this outcome (the fact) rather than the other (the foil)?'. PeCoX's perceptual XAI is model-agnostic, relying entirely on any learned model's input, output, and prospective feedback on that output. The ICM (or uncertainty) measures the expected performance of the machine learning model on any specific classification or decision. The perceptual XAI's diagnostic method to probe the agent behavior is thus purely reactive.

PeCoX's cognitive XAI selects goals, beliefs, and 'emotions' to explain why the agent chooses a certain action. This is similar to the BDI explanation models [85]. The model considers explanations from the intentional stance [55], i.e., the notion that an agent's action depends on its built-in intention, beliefs, or goals. PeCoX XAI includes another cognitive function: 'emotion', e.g., 'I expect (emotion) you would exercise regularly since I want (goal) you to be physically fit, and I suspect (belief) you are currently not exercising adequately' [56, 83]. PeCoX's cognitive framework is designed to be domain-agnostic. The cognitive XAI relies on Ontology Design Patterns for

reasoning and communication, and Interaction Design Patterns for shaping multimodal communications. Therefore, the cognitive XAI qualifies as a deliberative XGDAI.

3.4 Explanation Generation Techniques In Application Scenarios

To summarize, explanation generation techniques are broadly categorized as Deliberative if the agent intentionally develops a plan based on the current environmental states, Reactive if the agent changes its behavior in response to environmental stimuli, or Hybrid if the agent has elements of both Deliberative and Reactive. This section summarizes some of the explanation generation techniques used in the application scenarios.

Both Deliberative and Reactive explanation generation techniques are used in robot-human collaborative tasks. Deliberative techniques such as the Explainable-Plan technique provide the user with reasoning for the robot's planned methods to which the user can modify the robot's plans as needed [21, 118]. Other Deliberative techniques used in robot-human collaboration include CRE-ATIVE [219] for designing tools to meet expectations and Proactive [73] to generate explanations in anticipation of unexpected events. APE [88] is one example of a Reactive technique used in robot-human collaboration, while PeCoX [157] is considered a Hybrid technique.

The Explainable NPC [151] and GDA [152] are both Deliberative methods in game applications. The former is used to explain agents' actions and motivations within a game context. In contrast, the latter is used to explain unexpected events when observed states do not meet expectations. ARG [59], on the other hand, is a Reactive technique for producing human-like rationalization for agents playing a game.

Techniques used in training applications include Explainable BDI [85] and Debrief [102]. In both techniques, the mental states and justifications for selecting an action are stored in a log to be replayed if a similar scenario emerges. Explainable BDI produces explanations in the form of the beliefs and goals of the agent making the decision. At the same time, Debrief records the verbal assessments of a human operator making the decisions. The KAGR technique used in MAS [183] formulates agent reasoning for specific events as a quadruplet of Knowledge, Action, Goal, and Relation.

4 EXPLANATION COMMUNICATION TECHNIQUES FOR XGDAI

This section discusses what type of explanations (e.g., logs, text) are communicated from the agents to users and how they are communicated (e.g., visual, textual, speech). This section distinguishes explanation communication for XGDAI into visual, verbal, gestural, numerical, and textual expressions. Our findings are summarized in Table 4 in the Supplementary Materials.

4.1 Visualization

Visualization is a technique of communicating an agent's plan, "mind", or decision-making system by externalizing the pathways of the decision support process. This technique builds visual mediums between the agent and the humans to establish trust and transparency between humans and machines. An important justification for plan visualization, for instance, is the need to minimize the time taken to communicate the agents' plans in natural language to the humans in the loop. In some domains, visual explanations are well suited to be integrated into the typical workflow of experts: for example, in the medical domain, experts are used to analyze the results of imaging procedures (X-ray, fMRI, etc.), drawing their attention to potentially relevant parts of the image can be an intuitive interface.

Visualization techniques in this section are further divided into NN approaches and symbolic approaches. NN approaches provide low-level explainability of a model's internal representation using visual cues and, otherwise, leave interpretation up to the user. In symbolic approaches,

knowledge is encapsulated into a user-interpretable format, i.e., symbols directly computed using traditional connectionist models.

4.1.1 Neural Network Approaches. **Class Activation Map (CAM).** Deep learning architectures based on **convolutional neural networks (CNN)** achieved optimal results in many computer vision applications. From the training data, CNNs learn to extract a deep hierarchy of task-relevant visual features [10, 52, 117, 204]. While these feature-extracting filters can be visualized, they are hard to interpret: in lower layers, the filters are mostly edge detectors, while in higher layers, they are sensitive to complex features. Moreover, the filters represent what image features the CNN is sensitive to, but not what features lead to a given image classification. CAMs [230] address this issue by creating a heatmap of discriminative image regions over the input image that shows what parts of the input image contributed how much to the CNN's classification of the image as belonging to a selected class. In this way, CAMs supply a visual explanation for a classification. Furthermore, by calculating and comparing CAMs for different classes, relevant portions of the input image can be visualized for distinction.

CAMs work by inserting a **global average pooling (GAP)** layer directly after the architecture's final convolutional layer. This layer computes a spatial average of all filters from the previous layer, which are then weighted by the selected output class. A drawback of this approach is that the NN architecture is altered, and models need to be retrained. **Gradient-weighted Class Activation Mappings (Grad-CAM)** were introduced by Selvaraju et al. [186] based on gradients flowing into the final convolutional layer to solve this problem. They extend their approach further by fusing it with **Guided Backpropagation (Guided Grad-CAM)** to enhance the resolution. An example application field for CAM and related methods is medical image analysis. Ng et al. [158] use a three-dimensional CNN to analyze MRI data of possible migraine patients. They use CAMs to highlight discriminative brain areas. In such applications, CAMs guide a medical expert's attention to different image regions to assess the network's prediction.

Kerzel et al. [109] use Grad-CAM as a means for explaining neural robot grasping. Given an object to be grasped, a motor control network generates a movement command for a robot arm to pick up the object. Grad-CAM is then used to highlight the parts of the object which are relevant for generating the movement for a particular joint motor.

Attention Weights. **Recurrent neural networks (RNNs)** are a popular deep learning framework for temporal modeling, particularly in medical applications for modeling disease progression [62] and predicting diagnoses based on patient records [136]. One solution to improve interpretability or visualization was introducing an intermediate attention layer [40, 188]. The attention mechanism decomposes a complex input into a series of accumulated attention weights. For example, the Timeline model [12] aggregates contextual information of patient medical codes using an attention mechanism to predict future diagnoses of future medical visits. Medical practitioners were able to gain insight and anticipate future diagnoses by observing the attentional weights of medical codes over time. Theoretical scenarios can be simulated by manipulating attention weights and observing the subsequent prediction changes using attention-weight visualization tools [128, 138].

For instance, **neural machine translation (NMT)** uses tree search to translate a sentence from one language to another [128]. A search decoder determines which candidate words are the most appropriate translation based on the attention weights for specific words. A candidate-translated sentence is therefore generated by selecting specific candidate words that maximize the overall attention weights. The authors provided a visualization tool where the user can manually select other candidate words, to which the system recomputes the attention weights for subsequent

candidate words to complete the sentence. Color intensity is used for highlighting words with the highest attention, making it easier for users to observe the entire candidate sentence.

SHapley Additive exPlanation (SHAP). Lundberg and Lee [141] utilize SHapley Additive exPlanation (SHAP) values, an extension of Shapley values, as a united strategy to explain any machine learning model output. Given a collection of data points for training a prediction model, SHAP values are computed to determine the impact of a data point or feature on the model's predictions, giving the cumulative marginal contributions of individual inputs and features. SHAP values can thus be considered as a set of "importance scores". As the SHAP values are discretely computed for inputs and features, approximations of global input importance, or the global importance of a feature, can be easily calculated. Feature importance can be evaluated globally by taking the combined SHAP scores for a particular attribute or interaction with other features using a dependence scatter plot. The discrete SHAP values can be easily visualized, as shown with examples in Figure 11, providing users with easily interpretable information. The top sub-figure highlights the relation of the feature values (i.e., the color of the dots) and the impact of their contributions towards the model output based on their positions on the horizontal axis. The bottom sub-figure represents an example of an interaction plot between two features with each dot representing one data point. The combination of dot colors and their positions on the x-y axes forms an illustrative representation of the relation between the two features.

4.1.2 Neural-symbolic Approaches. Representations in NN are distributed among many neurons, making them difficult to isolate and interpret without relying on post-hoc knowledge extraction methods. Symbolic models such as graphs and decision trees are easier to interpret, as each node or symbol represents a single distinct concept. **Neural-symbolic (NS)** models combine the efficient learning and inference power of traditional NN and the interpretability of symbolic knowledge extraction and reasoning [70, 204].

Visual Grounding via NS-structured Graphs. Visual grounding is the general task of locating a structured description's components in an image [43]. The structured description or query can be a natural-language phrase that has been parsed as a symbolic structured graph such as a scene graph or action graph. In **human-robot interaction (HRI)** scenarios, visual grounding can enable a robot to accurately locate an object amongst other objects in the workspace [191]. Scene graphs were first used for the visual-grounding challenge by Johnson et al. [101]. The scene graph captures the visual scene's intricate semantics by expressly modeling objects, their descriptive properties, and their interconnections in the scene. For example, a scene described by the sentence "man riding black horse" is converted to a scene-graph representation with nodes corresponding to objects ("man"), descriptions or attributes ("horse is black"), and interconnections or relationships ("man riding horse"). The grounding task creates bounding boxes corresponding to the specified objects, such that the located objects have the specified attributes and relationships.

An NS "end-to-end" approach to visual grounding was proposed by Xu et al. [224] to generate structured scene representations from input images. The model addresses scene graph inference by using conventional RNNs that learn to generate image-grounded scene graphs and iteratively improve their predictions via message passing. A **Region Proposal Network (RPN)** [175] is used to generate a collection of object proposals from an image and transfers to the graph-inference module the extracted features of object regions. It then outputs a scene graph consisting of categories of objects, semantic relations between object pairs, and their bounding boxes. A significant aspect of the work is the message passing of contextual information between two sub-scene graphs (bipartite graphs) and how RNNs can refine their predictions.

In the same context, Yang et al. [226] present Graph R-CNN, a model for constructing scene graphs that can also perform object detection and infer object relations in images. The main
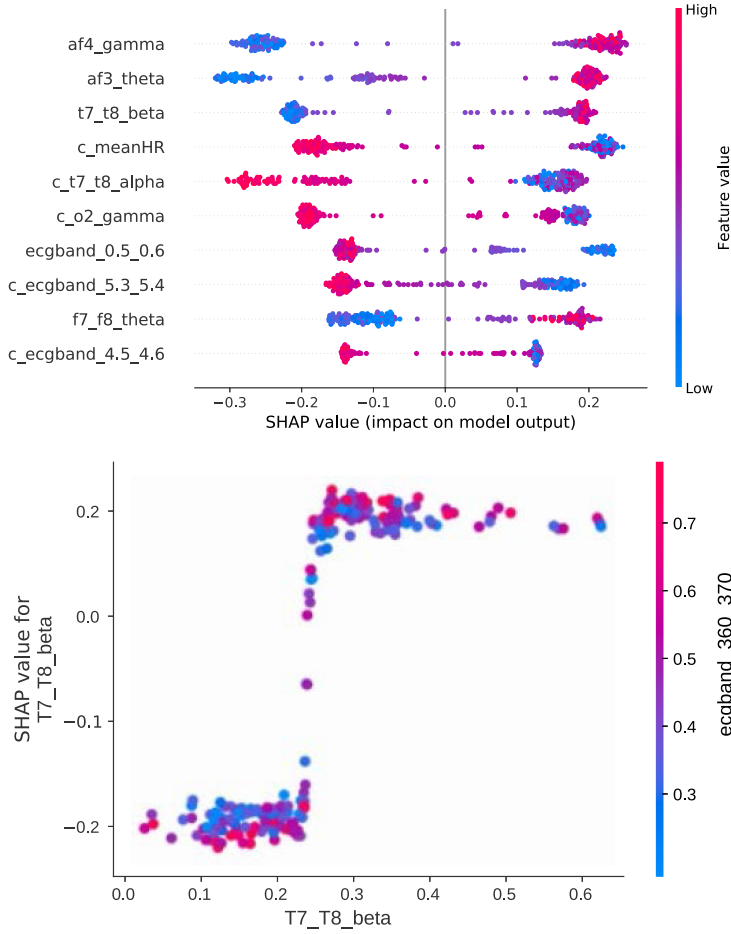
Fig. 11. Example of a SHAP summary plot (top) and feature dependence plot (bottom) describing feature importance for affect recognition [133].

framework consists of a **Relation Proposal Network (RePN)** dealing with the quadratic number of possible relationships between the objects and an **attentional Graph Convolutional Network (aGCN)**, which extracts context information between objects and their relations.

Zhou et al. [231] propose a weakly-supervised framework for reference-aware visual grounding for instructional videos. The only thing supervised is the temporal synchronization between the video segment and the transcription. The framework consists of a visually grounded action graph, i.e., an optimized **reference-aware multiple instance learning (RA-MIL)** objective for poor control of grounding in videos. A standardized representation captures the latent dependence between grounding variables and references in videos. Scene graph or action graph representations hold tremendous promise in visual grounding tasks, particularly for enhancing an agent's perceptual-cognitive framework. However, extracting scene graphs from images is still a significant challenge.

Extraction into Automata and Transducers. **Finite-state machines or automata (FSAs)** are computational models that assume one of any number of finite states at any time and transition

to other states in response to particular inputs. Architectures of FSAs are inherently interpretable and illustrated by a series of states linked by transition functions. A large body of work exists for extracting interpretable FSAs from RNNs. For example, using quantization algorithms [75], clustering outputs to infer FSAs [162, 229], and recursively testing inputs and outputs [215].

Transducers are FSAs with both an input and an output, whereas traditional FSAs have only an input. Transducers can be further divided into Moore machines and Mealy machines. The former produces outputs based solely on the machine state, while the latter relies on both inputs and states. Preference Moore machines [216] can operate as either a simple RNN or as a symbolic transducer while integrating different neural and symbolic knowledge sources. In a text-mining application [9], symbolic transducers were able to encode concepts from a news corpus quickly. At the same time, neural machines were able to produce improved classification performance with additional training. Transducer extraction [217] was then able to obtain detailed symbolic information while preserving sequential information.

Rule Extraction Algorithms. Rule extraction algorithms are a method for converting a neural network's internal arrangement of neurons into a linguistically interpretable format. The simplest form of a rule can be described using a conditional "IF-THEN" statement, represented in a NN by a single neuron as "IF input = X, THEN output = Y." Andrews et al. [7] classify rule extraction algorithms into three categories. Rule extraction algorithms that work at the neuron-level instead of the whole NN level or global are called decompositional methods. If the NN is a black-box model (i.e., activation or weights of the network cannot be observed directly), then the algorithms are pedagogical. The third category, eclectics, is a combination of decompositional and pedagogical algorithms.

The **Local Rule EXtraction (LREX)** method [147] is an example of eclectical rule extraction from **radial basis function (RBF)** networks. LREX examines the hidden unit to class assignments and extracts rules based on the input space to output space mappings. An example of decompositional rule extraction is the technique proposed by Malone et al. [144] which extracts propositional "IF-THEN" type rules from self-organized mapping networks.

Another class of rule extraction bypasses the need for neural networks and directly converts entire data sets into a more condensed set of rules. McGarry et al. [146] develop a method for automatically generating and integrating biological ontology into Bayesian network frameworks to enable reasoning and prediction. Mao et al. [145] introduce an algorithmic methodology for encoding relational representations of visual inputs in the form of sparse knowledge graphs. Sutherland et al. [199] introduce a context-agnostic methodology for extracting target-sentiment-cause triplets from sentences.

Given many data samples, extracting interpretable rules is a balancing act. Simplifying rules by discarding outliers may produce an oversimplified rule set that may not respond correctly to rare events. Sovrano et al. [195] propose a concept called **"eXplanation Awareness" (XA)** for organizing acquired experiences into clusters labeled on a case-by-case basis. XA enables sampling experiences in a curricular and task-oriented manner, focusing on events' rarity, importance, and meaning.

4.2 Textual And Verbal Communication

Textual Data Logs. For agents that rely on relational (or symbolic) logics for **knowledge representation and reasoning (KRR)**, using logs of data comes easily to communicate the internal state of the robots. An example is the explainable BDI model proposed by Harbers et al. [85], where all past mental states and actions of the agent needed for explanations are stored in a behavior log. When requesting an explanation, the algorithm selects beliefs and goals from the logs for the ex-

planation. Other examples are the XAI architecture proposed by Van Lent et al. [208] for NPCs in Full Spectrum Command, which extracts main events and decisions from the replay log during the post-review process.

Programming in Logic (Prolog). Prolog is a general-purpose logic programming language. An important aspect of this language is the utilization of symbolic representations - e.g., "?-dog(jane). no" corresponding to "Is Jane a dog? No - a cat". The Prolog program uses rules and facts and queries that search through stored facts and rules to work out answers [24]. Wicaksono and Sheh [219] use Prolog for the explainable CREATIVE model to express and store relevant information needed for explanations, such as acquired hypotheses, primitive object poses, and camera snapshots. CREATIVE represents a set of simple questions and answers and describes the relationship between objects as Prolog facts to enable explanation.

Verbal Communication. Speech or verbalization has been one of the earliest means of communicating agents' thoughts, beliefs, or actions, especially in the domain of social or service robots. Since social robots must converse daily with humans, they require skills for a natural conversation ability. Speech offers a more intuitive and faster means of explanation communication (compared to text) for robots, particularly interacting with the visually impaired or children who have not yet developed a full reading competency. The earliest generation of humanoid robots (i.e., developed in the 1970s and 1980s) was equipped with conversational skills, although the skills were usually primitive. They were typically designed as simplistic input/output mapping speech combinations [27]. In a study conducted by Ambsdorf et al. [4], participants' perceptions towards robots were evaluated as they watched two NICO robots playing tic-tac-toe against each other. While both robots utilized verbal communication, one robot only announced its moves while the other robot explained its reasonings for choosing a move. The explaining robot was perceived to be more human-like and lively compared to the non-explaining robot, which suggests that verbal explanations are a viable method to foster trust in human-robot cooperation tasks.

Another good example is the **Waseda Robot, WABOT-1** [68], which was designed with the capability to recognize spoken sentences as concatenated words, make vocal responses, and change a related state using a **Speech Input-Output System (SPIO)** [190]. The WABOT-1 system could accept Japanese spoken command sentences only in sequences of individually spoken Japanese words and then respond to the meaning of the command verbally to make the robot move as commanded. The system's inner core works as an automaton, making output and transitions after recognizing an input sentence. A further upgrade on the conversational system to make the speech more natural, particularly the speech synthesis part, was introduced in WABOT-2 [114], a robot musician which could produce speech responses by retrieving a word dictionary corresponding to a code of the spoken command. The dictionary stores the names of consonant and vowel syllabic units necessary for each unit's words, vowel durations, and accent patterns.

In this class, another more recent social robot is PaPeRo [163], a childcare robot designed to converse with humans more naturally. PaPeRo uses a dictionary of commonly used words and phrases and could also be updated by the designers. Humans interacting with the robot must also converse in similar words and phrases that the robot understands to enable natural conversation. PaPeRo uses an electronic hardware auditory system for human-robot communication [182] and can recognize multiple utterances, give a quiz to children who provide answers to the quiz using a special microphone and tell in natural language the names of the children who got the correct answer. Other similar robots are Honda Asimo [180] which uses a commercial hardware electronic system for speech synthesis, and ASKA [160] receptionist robots, with a conversational speech dialogue system that can recognize users' questions and answer the users' questions by a text-to-speech voice processing with other additional intuitive channels such as a hand gesture, head movement,

and body posture. Robovie [178] is another example in this category that can communicate in English using a vocabulary of about 300 sentences for speaking and 50 words for recognition. The reader is referred to Leite et al. [129] for a more comprehensive review of social robots.

Referring Expression. Referring expression is a language construction or expression used to identify particular objects within a scene [228]. **Generation of Referring Expressions (GRE)** or **Referring Expression Generation (REG)** is expression selection for identifying an element from a set of possible elements when given a shared context between a speaker and the hearer [67]. REG algorithms have traditionally been applied as computational models of people's ability to refer to objects [54]. Many scholarly works on REG can be found spanning the last five decades [119]. In recent years, REG has received significant attention in HRI scenarios, particularly for situated human-robot dialogues. In situated dialogue, humans and robots/agents are co-present in a shared physical environment or workspace (e.g., in assembly tasks) and often need to refer to an object in the environment that is being shared or manipulated [63]. The motivation for REG is thus the need for the robot to comprehend expressions concerning objects in a shared space and their relationships from inputs of image and natural language. In a contribution by Giuliani et al. [77], a situated reference generation framework is presented for an HRI system to collaborate with humans in an assembling task that involves building objects from a set of wooden toys. The REG system includes a goal inference sub-symbolic system that can identify people's goals or mistakes by observing their verbal and non-verbal actions. The robot's REG module then uses situated references to clarify to users the mistakes and suggest strategies for solutions [50].

Another relevant example is the **INteractive visual Grounding of Referring ExpreSSions (INGRESS)** framework proposed by Shridhar et al. [191]. INGRESS is a robot that executes instructions to pick and place items (in the environment) in natural language. The system can ground referring expressions, i.e., recognizing dialogues or references to objects and their relationship from images and natural language inputs. It may also ask questions for interactive clarification of unclear referential expressions. It uses a two-stage **long short-term memory (LSTM)** framework to establish visual representations of objects and link them to input language expressions to identify the candidate referred-to-objects.

Some other relevant contributions are the gestural-deictic reference by Kranstedt and Wachsmuth [120] and the haptic-ostensive reference by Foster et al. [67], in which the authors argue that a multimodal reference involving manipulation actions with the objects would provide a richer referring approach for conversational partners in situated dialogue. They suggested that a fully elaborated linguistic reference may not always be necessary between conversational partners.

4.3 Explanation Communication Techniques In Application Scenarios

In summary, explanation communication techniques are grouped by explanatory forms: visual, numerical, verbal, and textual. This section summarizes some of the explanation communication techniques used in application scenarios. Communication techniques used in robot-human collaborative tasks mainly include human-like expressive abilities such as gestures [148], motion [148], and verbal communication [114, 160, 163, 178, 228]. The familiarity of human-like communication is more intuitive and user-friendly for people new to interacting with intelligent robots. Similarly, simple communication methods such as expressive lights [13, 194] are unexplored territory for XGDAI.

One domain where AIs are highly in demand is e-health for applications such as AI-assisted medical diagnosis caretaker robots. Visualization methods such as CAM [158] highlight regions of interest that are relevant to the diagnosis. Numerical methods such as attention weights

[40, 136, 188] are used for fine-tuning diagnosis algorithms to minimize misdiagnoses. In scenarios where AIs are used for training scenarios, XGDAI helps to diagnose how and why the agent makes a particular decision during field exercises. Logged recordings of the agents' actions and their rationales [85, 208] can then be parsed to identify and correct any behavior that may produce undesirable outcomes.

5   CONTINUAL LEARNING FOR EXPLAINABILITY

In this section, we examine techniques in XGDAI that enable continual learning of explanatory knowledge, domain knowledge, domain models, or policies (e.g., sets of environment states) for explanation generation. This section explores the solution to two issues: handcrafting domain knowledge artifacts for explanation generation in deliberative symbolic agents; and learned policy losses during the decision-making process for explainability in reactive RL agents.

5.1   Case-Based Reasoning (CBR)

For agents that rely on handcrafted domain knowledge for defining the explanation components (e.g., expectations, discrepancy in definitions, knowledge of how to resolve the discrepancy, operator feedback), the common challenge is that substantial domain engineering is done on the system which needs to be updated each time the robot changes to a new environment. CBR is used in many XGDAI approaches to acquire domain knowledge at run-time to minimize the prerequisite domain engineering for generating explanations. CBR applies a learning and adaptation strategy to help explanation generation by storing, recalling, and adapting knowledge information or experiences (or cases) stored in a case-library or memory [205].

In the CBR framework, agent learning to expand its knowledge is executed by evaluating and integrating new experiences into the case-library and/or re-indexing and reusing previous experiences [116]. CBR has two primary learning methods: observational learning (supervised) and learning through personal experience [205]. Observational learning is performed by filling the case-library with observations from expert demonstrations or actual data [209]. Learning from one's personal experience [115] happens after a reasoning process that evaluates a potential solution to a challenge. If the solution succeeds, it is saved and applied for future references [205].

Weber et al. [214] propose an observational learning approach to reduce the number of domain knowledge acquisitions necessary to implement the real-time strategy game StarCraft. CBR was applied to learn expectations, explanations, and goals from expert demonstrations. An adversary library offers examples of adversary behavior, and a goal library chooses goals to be pursued by the agent. The learning system offers explanations to the agent if it achieves an intended goal or if a deviation is observed. In a similar effort, Floyd and Aha [66] apply two CBR systems to assist agent learning and explanation generation. Both CBRs use cases learned while interacting with the operator, i.e., learning by observation. The first CBR process determines if a robot is trustworthy and selects a new behavior if found otherwise. The second CBR is used to provide an explanation or clarification should the robot change its behavior. The agent's explanations are based on explicit feedback received from an operator. The model is evaluated in a simulation environment involving an operator who instructs the robot to patrol, detect suspicious objects, and designate the objects as either harmless or dangerous.

5.2   Explainable Reinforcement Learning

One of the major concerns for many RL agents is a loss of information about the decision process during the agent's policy learning process. An RL agent may know at the end of a learning objective that certain actions may produce a higher gain or value to attain the goal or that one action is preferred over another but loses the rationale behind the decision-making as the policy
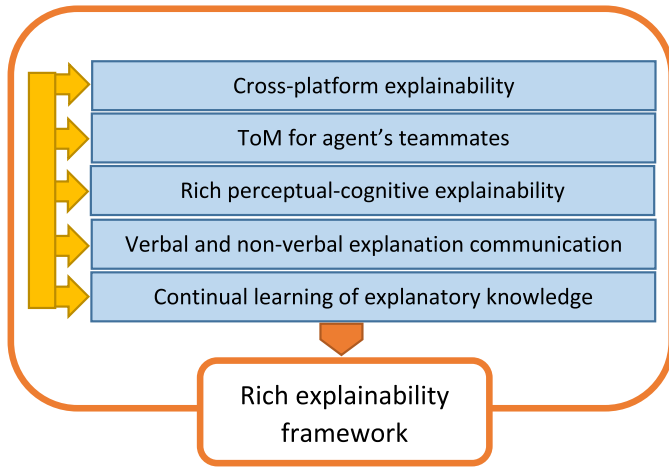
Fig. 12. Explainability Framework for XGDAI.

converges towards an optimal mechanism for the selection of actions [187]. Once an optimal policy has been learned, the lack of bookkeeping, traceability, or recovery of this process makes it hard for the agent to explain itself or transfer a learned policy for explainability. A few extensions to applying the XRL technique seek to enhance retention of the learned policy for explaining agents' decision processes. Some examples include MXRL proposed by Cruz et al. [47] which introduces an episodic memory to store important events during the robot's decision-making process. The episodic memory enables the agent to introspect, observe, or analyze its environment transitions and interactions. The agent is shown to clarify its actions to lay users during task execution, relying on its episodic memory. The major shortcomings in the MXRL technique, as with many other memory-based continual learning techniques, are the limitations with regard to the use of memory in large solution spaces. There is still an open-ended quest for XRL techniques that enable comprehensibility, continual learning, and policy retention.

6 DISCUSSION

Existing studies in XGDAI show a lack of consensus on the requirements for explainability. Different behavioral architectures for GDAI - e.g., deliberative, reactive, and hybrid - come with different explanation generation techniques. The state of the art suggests the need for an effective unified approach towards explainability in XGDAI. Many explainability techniques still lack an extensive framework consisting of a rich perceptual-cognitive explainable framework, a verbal and non-verbal communication framework, a framework for natural language processing, and continual learning for explanation construction. In this section, we outline a framework, illustrated in Figure 12, for the effective actualization of explainability in XGDAI and a road map in Figure 13 illustrating the four phases of explanation in future research.

6.1 Framework For Explainability In XGDAI

6.1.1 Cross-platform Explainability. In the current state of the art, explanation techniques, particularly for symbolic deliberative XGDAI, are domain-specific, relying heavily on the domain knowledge (or learning of the domain knowledge) for constructing explanations. In many such applications, agents already have plans and a clear decision system to achieve a goal; thus, constructing a rich explanation is often relatively straightforward. However, applying these techniques can
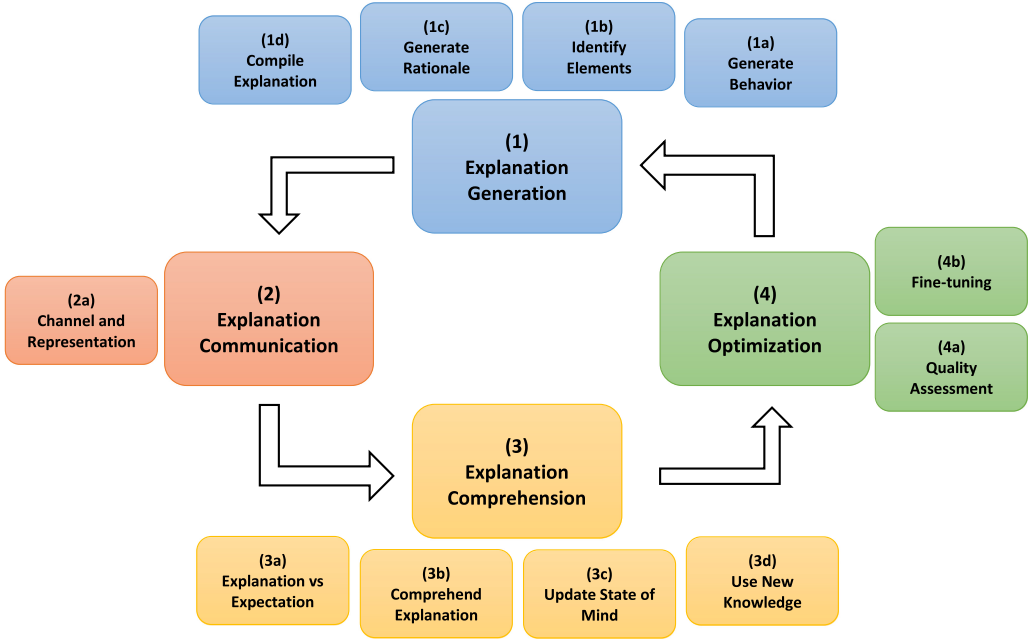
Fig. 13. Four phases of explanation. The explanation cycle begins by generating an explanation (1). The algorithms' behavior determines when an explanation should be generated (1a), followed by identifying the elements needed to construct the explanation (1b). The explanation and the rationale are then compiled (1c-1d). The channel and representation determine how the explanation is conveyed (2a). The recipient compares the received explanation against its expectations (3a) and attempts to fit the new information into its world model (3b-3c). Subsequent actions then take into account the new information (3d). The performance of the agent with the updated information is assessed compared to its previous performance (4a). Using optimization algorithms such as genetic algorithms, the effectiveness of the agent is incrementally improved (4b) by adjusting the explanation parameters in (1), (2), and (3).

be problematic as agents perform optimally in a specific domain and sub-optimal in other domains where knowledge of the agent's world is not fully represented in the framework. A cross-platform explainability framework can significantly benefit existing work to improve agents' performance and minimize re-engineering of the domain knowledge.

Some emerging techniques, such as the domain-agnostic approaches, suggest a useful notion of explainability that enables cross-platform explanation generation for agents and robots. Most of these techniques can be found for reactive black box agents whose decisions are based solely on the current environment state, not defined *a priori*. Without a world or domain knowledge model, these agents can generate explanations based on the policy learned. However, these platforms are less extensive, and a significant research effort is still required. A cross-platform explainability approach should significantly benefit both deliberative and reactive agents. Also beneficial would be approaches for bridging the gap between symbolic and reflexive (neural network) approaches to explanation generation and communication.

6.1.2 ***Theory of Mind (ToM)*** *for Agents' Teammates.* ToM tends to reason about other people's perceptions, beliefs, and goals and take them into account [30]. A significant body of work on XGDAI involves agents/robots collaborating with humans and other agents. Given this reality, it is imperative for agents to adequately understand their teammates for effective collaboration

and provide a useful and timely explanation when necessary. A useful step in this direction may be to integrate the ToM concept in the explanation framework enabling an agent to also reason about other teammates' perceptions and mental states. A well-constructed ToM should enable the robot to understand its teammates' expectations and provide a useful, relevant, and timely explanation. The motivation here is that humans are well known to collaborate with their teammates extensively and explain their behavior using the ToM concept. Currently, only one approach mentioned generating explanations about user/team-member expectations and possible violations of expectations [73].

6.1.3 Rich Perceptual-cognitive Explainability Framework. A significant number of approaches, particularly those on deliberative XGDAIs, explain at the level of agents' cognitive functions, which are not grounded on actual agents' perceptions of the real world. On the other hand, a few studies, mainly reactive XGDAIs, highlight procedures for explanation generation at the level of the agent's perceptual function (sensor information, environment states, etc.) with a poor or non-existent explainable cognitive framework or explainable decision-making framework. A rich perceptual-cognitive explainable framework that abstracts low-level agent perception (primarily perceptual explanatory knowledge) for high-level cognition and explainability would significantly advance the current research on XGDAI. Some emerging techniques of explainability/interpretability would be useful, e.g., CAMs to identify the salient information in an image, visual grounding to localize an image component, and REG to enable an agent to distinguish between objects in a shared workspace (e.g., in situated human-robot dialogues).

6.1.4 Natural Language Processing (NLP). NLP is the computer system or AI system's ability to understand, analyze, manipulate, and generate human language [97]. NLP is crucial in the explanation generation/communication framework for XGDAI to enable a human-comprehensible explanation of an agent's/robot's perception, cognition, or decisions. Currently, the state of the art in XGDAI reveals less extensive or even a non-existent NLP ability for many of the agents and robots surveyed. Adding a rich NLP system to existing frameworks would significantly benefit XGDAI in its usefulness and applicability.

6.1.5 Integrating Verbal and Non-verbal Explanation Communication. The current state of the art reveals different explanation communication modalities for XGDAI applied in separate niche areas/scenarios. With diverse application scenarios, the need to communicate the agent's/robot's plan, decision, intentions, and so on, by combining both verbal and non-verbal communication means is necessary if such an explanation shall be natural and effective. A relevant example is seen in how humans explain/communicate using verbal and non-verbal communication, depending on which is most effective for the circumstance. XGDAI should also enable such a combination of different modalities.

For example, for agents sharing a pedestrian walkway with humans, rotational head/eye movements seem more natural and effective to communicate the agent's decision to turn left, change paths, and so on. [148]. Simultaneously, verbal communication would also prove useful to alert other pedestrians of the agent approaching when they are not aware of its presence. For collaboration with teammates, speech for normal explanatory conversation is necessary. However, expressive motion like gesture (e.g., head nodding) would be useful to provide an implicit explanation, or an expressive light [13, 194] or sound to alert teammates of its mental state in an emergency. Kerzel et al. [109] propose a framework which integrates multiple types of explainability methods in a neuro-robotic architecture for object recognition and grasping: first, an LED-based facial expression on the robot's face is used to give a human interaction partner quick and intuitive feedback about understanding the given task (positive emotion expression) and possible issues

(negative emotion expression); second, a more detailed verbal explanation of the object recognition outcomes is generated; and third, a visual-based Grad-CAM is used to show the user how different areas of the scene, as perceived by the robot, correlate with specific joint motor movements while grasping the object. There can be many possible effective and natural combinations of explanation communication modalities. More research work is still required to bridge this gap.

6.1.6 Continual Learning of Explanatory Knowledge. As agents/robots interact with their environment, teammates, and supervisors, they are expected to explain their decisions or actions in different situations and scenarios. Handcrafting explanatory knowledge to satisfy all possible situations and expectations is difficult and requires significant effort or domain engineering. In this respect, an agent's ability to continuously learn to generate/construct explanations in different situations is crucial to explainable agents' success. As in traditional machine learning, learning could be achieved by supervised learning (e.g., learning from demonstration [209]), unsupervised learning [115], and reinforcement learning. CBR [205], and XRL [47] techniques have been applied in a few studies on deliberative and reactive agents respectively to address this concern. However, for reactive XRL, a major concern is how to retain policies learned by agents for constructing explanations. There is still a significant gap to fill in this direction. Scalability and resource management issues would also need to be addressed if explanatory knowledge is stored in a continual learning framework.

6.2 Road Map
This section proposes a road map for extending research into XGDAI, based on the representation shown in Figure 13. The explanation cycle is categorized as the four main phases Explanation Generation, Explanation Communication, Explanation Comprehension, and Explanation Optimization.

6.2.1 Explanation Generation. This phase concerns the elements needed to activate and generate explanations. The explanation generation phase is dependent on the behavior of the AI of the agent or robot. The key research directions of this phase are as follows:

(1a) Existing works focus on advancing the capabilities of the agent (i.e., cognitive and behavioral architecture). However, most methods do not support explainability functions. To further research into XGDAI, equipping the agent's internal AI architecture with a mechanism to generate explanations is needed.

(1b)–(1d) Although Kaptein et al. [105] describe how context-awareness and personalization are essential for XGDAI, few works in the literature fulfill the criteria. There is a need for new techniques to facilitate (1b) identification of elements relevant for an explanation; (1c) determining the explanation rationale; and (1d) assembling the elements into a sound explanation.

6.2.2 Explanation Communication. This phase is needed to convey the generated explanation in an understandable format to the other party, which can be another agent or a user. The identified research directions are:

(2a) Explainable agents are likely to be deployed in diverse environments. Therefore, there is a need for the agent to support more than one type of communication channels (i.e., visual, auditory, expressive). Approaches in this direction are rarely found in the literature even though this is an essential element to effectively and efficiently convey explainable information. Although depending on the application scenario, comprehensive explanations can be conveyed using simple communication channels such as expressive lighting signals [13, 194].

XGDAI	Generate				Communicate	Comprehend				Optimize	
	(1a)	(1b)	(1c)	(1d)	(2a)	(3a)	(3b)	(3c)	(3d)	(4a)	(4b)
GDA	■	■	■	■		■	■	■	■	■	
Explainable BDI	■	■	■	■				■			
SAT	■	■	■			■					
Meta-AQUA	■	■	■	■	■	■	■	■	■	■	■
Proactive	■	■			■						
KAGR	■	■	■	■							
XAI-Plan	■	■	■	■							
Explainable NPC	■	■	■			■		■		■	
Debrief	■	■	■	■				■			■
CREATIVE	■	■				■	■	■		■	
ARG	■	■	■	■		■	■	■	■	■	■
MXRL	■					■	■	■	■	■	■
MSX		■				■	■	■	■	■	■
APE		■	■	■		■					
PeCoX	■	■	■				■				

Fig. 14. Reviewed XGDAI techniques mapped to the four proposed phases of explanation.

6.2.3 Explanation Comprehension. In this phase, the receiver receives the generated explanation through the chosen communication channel to be converted into useful information. The following points should be considered:

(3a)–(3b) Metrics should measure the efficiency of the generated explanation as well as the usefulness of the explanation to the receiver.

(3c)–(3d) The agent/robot keeps track of a model of the user knowledge. The model is regularly updated to reflect the growth of the user expertise and how the user views the **State of Mind (SoM)** of the agent/robot.

6.2.4 Explanation Optimization. Completing the explanation cycle, this phase tracks the explanation flow from beginning to end to identify elements that can be fine-tuned to improve the quality of future explanations. To that end, the following research directions are identified:

(4a) Several elements in the explanation cycle can be changed to improve the effectiveness of the explanations. For example, depending on the user feedback in (3), explanation elements in (1) can be altered to exclude redundant information. In addition, the user can query for more information leading to a different choice of explanation generation technique in (1d) or choice of communication medium in (2). The modifications are then evaluated to determine if the quality of the explanations has improved.

(4b) Based on the compiled information in (4a), an optimization algorithm such as evolutionary algorithms or reinforcement learning is used to modify elements in the explanation cycle to improve the effectiveness of future explanations.

Figure 14 shows how the reviewed XGDAI strategies in this work were mapped to the four proposed phases of explanation. As expected, a majority possessed elements for generating explanations while fewer have the ability to assess user comprehension and further fine-tune the explanation generation. Explanation communication, however, was a rarely explored domain. Meta-AQUA and Proactive were two methods that actively determined the communication channel as a viable technique to facilitate explanations. Future research directions may further explore the use of multimodal communication to improve the effectiveness of explainability methods.

7 CONCLUSIONS AND FURTHER WORK

The field of XGDAI is emerging with many rich applications in several domains. Explainability enables transparency for these agents/robots and encourages users' trust for applications in safety-critical situations. This survey presents several techniques for explanation generation and communication proposed and implemented in XGDAI until today. Typically, many XGDAIs techniques can enable the robots/agents to justify and explain their decisions and plan rationales for their actions. However, the state of the art shows that current approaches on XGDAIs are still in their infancy, lacking an extensive explanation generation and communication framework. Consequently, this study highlights a framework towards a more extensive XGDAI that has an extended perceptual and cognitive explanation capability. Future work will involve developing a transparent and domain-agnostic integrated architecture for the effective actualization of explainability in XGDAI.

A SUPPLEMENTARY MATERIALS

Supplementary materials are available in the online version of this paper.

REFERENCES

[1] Henny Admoni and Brian Scassellati. 2017. Social eye gaze in human-robot interaction: A review. *Journal of Human-Robot Interaction* 6, 1 (2017), 25–63.

[2] Fahed Alkhabbas, Romina Spalazzese, and Paul Davidsson. 2020. An agent-based approach to realize emergent configurations in the Internet of Things. *Electronics* 9, 9 (2020), 1347.

[3] Ahmed Alqaraawi, Martin Schuessler, Philipp Weiß, Enrico Costanza, and Nadia Berthouze. 2020. Evaluating saliency map explanations for convolutional neural networks: A user study. In *Proceedings of the 25th International Conference on Intelligent User Interfaces*. Association for Computing Machinery, New York, NY, United States, 275–285.

[4] Jakob Ambsdorf, Alina Munir, Yiyao Wei, Klaas Degkwitz, Harm Matthias Harms, Susanne Stannek, Kyra Ahrens, Dennis Becker, Erik Strahl, Tom Weber, et al. 2022. Explain yourself! Effects of Explanations in Human-Robot Interaction. 8 pages. https://doi.org/10.48550/ARXIV.2204.04501

[5] Dan Amir and Ofra Amir. 2018. Highlights: Summarizing agent behavior to people. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, United States, 1168–1176.

[6] Jacob Andreas, Anca Dragan, and Dan Klein. 2017. Translating Neuralese. 14 pages. https://doi.org/10.48550/ARXIV.1704.06960

[7] Robert Andrews, Joachim Diederich, and Alan B. Tickle. 1995. Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-based Systems* 8, 6 (1995), 373–389.

[8] 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 MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, United States, 1078–1088.

[9] Garen Arevian, Stefan Wermter, and Christo Panchev. 2003. Symbolic state transducers and recurrent neural preference machines for text mining. *International Journal of Approximate Reasoning* 32, 2–3 (2003), 237–258.

[10] Sercan O. Arik and Yu-Han Liu. 2020. Explaining Deep Neural Networks Using Unsupervised Clustering. 13 pages. https://doi.org/10.48550/ARXIV.2007.07477

[11] Bryan Auslander, Matthew Molineaux, David W. Aha, Allen Munro, and Quentin Pizzini. 2009. *Towards Research on Goal Reasoning with the TAO Sandbox*. Technical Report. Navy Center for Applied Research in Artificial Intelligence Washington DC.

[12] Tian Bai, Shanshan Zhang, Brian L. Egleston, and Slobodan Vucetic. 2018. Interpretable representation learning for healthcare via capturing disease progression through time. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. Association for Computing Machinery, New York, NY, United States, 43–51.

[13] Kim Baraka, Ana Paiva, and Manuela Veloso. 2016. Expressive lights for revealing mobile service robot state. In *Robot 2015: Second Iberian Robotics Conference*. Springer, Springer Cham, Switzerland, 107–119.

[14] Kevin Baum, Holger Hermanns, and Timo Speith. 2018. From machine ethics to machine explainability and back. In *International Symposium on Artificial Intelligence and Mathematics ISAIM2018*. ISAIM, Fort Lauderdale, Florida, United States, 8 pages.

[15] Guus Beckers, Joris Sijs, Jurriaan van Diggelen, Roelof JE van Dijk, Henri Bouma, Mathijs Lomme, Rutger Hommes, Fieke Hillerstrom, Jasper van der Waa, Anna van Velsen, et al. 2019. Intelligent autonomous vehicles with an extendable knowledge base under meaningful human control. In *Counterterrorism, Crime Fighting, Forensics, and Surveillance Technologies III*, Vol. 11166. International Society for Optics and Photonics, 111660C.

[16] Julien Bidot, Susanne Biundo, Tobias Heinroth, Wolfgang Minker, Florian Nothdurft, and Bernd Schattenberg. 2010. Verbal plan explanations for hybrid planning. In *Proceedings of Multikonferenz Wirtschaftsinformatik*. Citeseer, 2309–2320.

[17] Or Biran and Courtenay Cotton. 2017. Explanation and justification in machine learning: A survey. In *IJCAI-17 Workshop on Explainable AI (XAI)*, Vol. 8.

[18] Susanne Biundo and Bernd Schattenberg. 2001. From abstract crisis to concrete relief - a preliminary report on combining state abstraction and HTN planning. In *Proceedings of the 6th European Conference on Planning*. Springer Verlag, 157–168.

[19] Ned Block. 2005. Two neural correlates of consciousness. *Trends in Cognitive Sciences* 9, 2 (2005), 46–52.

[20] Gregory Bonaert, Youri Coppens, Denis Steckelmacher, and Ann Nowe. 2021. Explainable Reinforcement Learning Through Goal-Based Interpretability. https://openreview.net/forum?id=IlJbTsygaI6.

[21] Rita Borgo, Michael Cashmore, and Daniele Magazzeni. 2018. Towards providing explanations for AI planner decisions. 7 pages. https://doi.org/10.48550/ARXIV.1810.06338

[22] Mathieu Bourgais, Patrick Taillandier, and Laurent Vercouter. 2019. Ben: An agent architecture for explainable and expressive behavior in social simulation. In *International Workshop on Explainable, Transparent Autonomous Agents and Multi-Agent Systems*. Springer, 147–163.

[23] Michael W. Boyce, Jessie Y. C. Chen, Anthony R. Selkowitz, and Shan G. Lakhmani. 2015. Effects of agent transparency on operator trust. In *Proceedings of the 10th Annual ACM/IEEE International Conference on Human-Robot Interaction Extended Abstracts*. 179–180.

[24] Max Bramer. 2005. *Logic Programming with Prolog*. Vol. 9. Springer.

[25] Michael Bratman. 1987. *Intention, Plans, and Practical Reason*. Harvard University Press Cambridge, MA.

[26] Cynthia Breazeal, Cory D. Kidd, Andrea Lockerd Thomaz, Guy Hoffman, and Matt Berlin. 2005. Effects of nonverbal communication on efficiency and robustness in human-robot teamwork. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 708–713.

[27] Cynthia Breazeal, Atsuo Takanishi, and Tetsunori Kobayashi. 2008. *Social Robots that Interact with People*. Springer Berlin, 1349–1369.

[28] Joost Broekens, Maaike Harbers, Koen Hindriks, Karel Van Den Bosch, Catholijn Jonker, and John-Jules Meyer. 2010. Do you get it? User-evaluated explainable BDI agents. In *German Conference on Multiagent System Technologies*. Springer, 28–39.

[29] Alan Bundy. 2017. Preparing for the future of artificial intelligence. *AI & Society* 32, 2 (2017), 285–287.

[30] Lindsey Jacquelyn Byom and Bilge Mutlu. 2013. Theory of mind: Mechanisms, methods, and new directions. *Frontiers in Human Neuroscience* 7 (2013), 413.

[31] Peter Carey. 2018. *Data Protection: A Practical Guide to UK and EU Law*. Oxford University Press, Inc.

[32] Ravi Teja Chadalavada, Henrik Andreasson, Robert Krug, and Achim J. Lilienthal. 2015. That's on my mind! Robot to human intention communication through on-board projection on shared floor space. In *European Conference on Mobile Robots*. IEEE, 6 pages.

[33] Tathagata Chakraborti, Kshitij P. Fadnis, Kartik Talamadupula, Mishal Dholakia, Biplav Srivastava, Jeffrey O. Kephart, and Rachel K. E. Bellamy. 2017. Visualizations for an explainable planning agent.

[34] Tathagata Chakraborti, Sarath Sreedharan, Yu Zhang, and Subbarao Kambhampati. 2017. Plan explanations as model reconciliation: Moving beyond explanation as soliloquy.

[35] Arjun Chandrasekaran, Deshraj Yadav, Prithvijit Chattopadhyay, Viraj Prabhu, and Devi Parikh. 2017. It takes two to tango: Towards theory of AI's mind.

[36] Wang Chang, W. U. Lizhen, Yan Chao, Wang Zhichao, Long Han, and Yu Chao. 2020. Coactive design of explainable agent-based task planning and deep reinforcement learning for human-UAVs teamwork. *Chinese Journal of Aeronautics* 33, 11 (2020), 2930–2945.

[37] Joyjit Chatterjee and Nina Dethlefs. 2020. XAI4Wind: A Multimodal Knowledge Graph Database for Explainable Decision Support in Operations & Maintenance of Wind Turbines.

[38] Jessie Y. C. Chen, Shan G. Lakhmani, Kimberly Stowers, Anthony R. Selkowitz, Julia L. Wright, and Michael Barnes. 2018. Situation awareness-based agent transparency and human-autonomy teaming effectiveness. *Theoretical Issues In Ergonomics Science* 19, 3 (2018), 259–282.

[39] Jessie Y. Chen, Katelyn Procci, Michael Boyce, Julia Wright, Andre Garcia, and Michael Barnes. 2014. *Situation Awareness-based Agent Transparency*. Technical Report. Army Research Lab Aberdeen Proving Ground MD Human Research and Engineering Directorate.

[40] Edward Choi, Mohammad Taha Bahadori, Jimeng Sun, Joshua Kulas, Andy Schuetz, and Walter Stewart. 2016. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. In *Advances in Neural Information Processing Systems*. 3504–3512.

[41] Jaegul Choo and Shixia Liu. 2018. Visual analytics for explainable deep learning. *IEEE Computer Graphics and Applications* 38, 4 (2018), 84–92.

[42] Noel C. F. Codella, Michael Hind, Karthikeyan Natesan Ramamurthy, Murray Campbell, Amit Dhurandhar, Kush R. Varshney, Dennis Wei, and Aleksandra Mojsilovic. 2018. Teaching meaningful explanations.

[43] Erik Conser, Kennedy Hahn, Chandler M. Watson, and Melanie Mitchell. 2019. Revisiting Visual Grounding.

[44] Ruben Contreras, Angel Ayala, and Francisco Cruz. 2020. Unmanned aerial vehicle control through domain-based automatic speech recognition. *Computers* 9, 3 (2020), 75.

[45] Michael T. Cox. 2007. Perpetual self-aware cognitive agents. *AI Magazine* 28, 1 (2007), 32–32.

[46] Michael T. Cox and Ashwin Ram. 1999. *Introspective Multistrategy Learning: On the Construction of Learning Strategies*. Technical Report. Wright State Univ. Dayton OH Dept of Computer Science and Engineering.

[47] Francisco Cruz, Richard Dazeley, and Peter Vamplew. 2019. Memory-based explainable reinforcement learning. In *Australasian Joint Conference on Artificial Intelligence*. Springer, 66–77.

[48] Luca Cultrera, Lorenzo Seidenari, Federico Becattini, Pietro Pala, and Alberto Del Bimbo. 2020. Explaining autonomous driving by learning end-to-end visual attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 340–341.

[49] Mauro Da Lio, Riccardo Donà, Gastone Pietro Rosati Papini, and Kevin Gurney. 2020. Agent architecture for adaptive behaviors in autonomous driving. *IEEE Access* 8 (2020), 154906–154923.

[50] Robert Dale and Ehud Reiter. 1995. Computational interpretations of the Gricean maxims in the generation of referring expressions. *Cognitive Science* 19, 2 (1995), 233–263.

[51] Dustin Dannenhauer, Michael W. Floyd, Matthew Molineaux, and David W. Aha. 2018. Learning from exploration: Towards an explainable goal reasoning agent. In *Proceedings of the Workshop on Adaptive Learning Agents (held at the 27th International Joint Conference on Artificial Intelligence and the 23rd European Conference on Artificial Intelligence)*. 1–8.

[52] Arun Das and Paul Rad. 2020. Opportunities and challenges in explainable artificial intelligence (XAI): A survey.

[53] Paul Davidsson. 1996. *Autonomous Agents and the Concept of Concepts*. Department of Computer Science, Lund University.

[54] Kees van Deemter, Albert Gatt, Ielka van der Sluis, and Richard Power. 2012. Generation of referring expressions: Assessing the incremental algorithm. *Cognitive Science* 36, 5 (2012), 799–836.

[55] Daniel Clement Dennett. 1978. Three kinds of intentional psychology. *Perspectives in the Philosophy of Language: A Concise Anthology* (1978), 163–186.

[56] Sabine A. Döring. 2003. Explaining action by emotion. *The Philosophical Quarterly* 53, 211 (2003), 214–230.

[57] Anca D. Dragan, Shira Bauman, Jodi Forlizzi, and Siddhartha S. Srinivasa. 2015. Effects of robot motion on human-robot collaboration. In *10th ACM/IEEE International Conference on Human-Robot Interaction*. IEEE, 51–58.

[58] Upol Ehsan, Brent Harrison, Larry Chan, and Mark O. Riedl. 2018. Rationalization: A neural machine translation approach to generating natural language explanations. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*. 81–87.

[59] Upol Ehsan, Pradyumna Tambwekar, Larry Chan, Brent Harrison, and Mark O. Riedl. 2019. Automated rationale generation: A technique for explainable AI and its effects on human perceptions. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*. 263–274.

[60] Mica R. Endsley. 2018. *Innovative Model for Situation Awareness in Dynamic Defense Systems*. CRC Press.

[61] Susan L. Epstein, Anoop Aroor, Matthew Evanusa, Elizabeth I. Sklar, and Simon Parsons. 2015. Learning spatial models for navigation. In *International Conference on Spatial Information Theory*. Springer, 403–425.

[62] Cristóbal Esteban, Oliver Staeck, Stephan Baier, Yinchong Yang, and Volker Tresp. 2016. Predicting clinical events by combining static and dynamic information using recurrent neural networks. In *2016 IEEE International Conference on Healthcare Informatics*. IEEE, 93–101.

[63] Rui Fang, Changsong Liu, Lanbo She, and Joyce Chai. 2013. Towards situated dialogue: Revisiting referring expression generation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. 392–402.

[64] Richard E. Fikes and Nils J. Nilsson. 1971. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence* 2, 3–4 (1971), 189–208.

[65] Rhona Flin and Kevin Arbuthnot. 2017. *Incident Command: Tales From The Hot Seat*. Routledge.

[66] Michael W. Floyd and David W. Aha. 2016. Incorporating transparency during trust-guided behavior adaptation. In *International Conference on Case-Based Reasoning*. Springer, 124–138.

[67] Mary Ellen Foster, Ellen Gurman Bard, Markus Guhe, Robin L. Hill, Jon Oberlander, and Alois Knoll. 2008. The roles of haptic-ostensive referring expressions in cooperative, task-based human-robot dialogue. In *3rd ACM/IEEE International Conference on Human-Robot Interaction*. IEEE, 295–302.

[68] Hiromichi Fujisawa, Katsuhiko Shirai, and Yukio Sakai. 1973. Ear and voice of the WABOT - the speech-input-output system. *Bulletin of Science and Engineering Research Laboratory Waseda* (7 1973), 39–85.

[69] Xiaofeng Gao, Ran Gong, Yizhou Zhao, Shu Wang, Tianmin Shu, and Song-Chun Zhu. 2020. Joint mind modeling for explanation generation in complex human-robot collaborative tasks. In *29th IEEE International Conference on Robot and Human Interactive Communication*. IEEE, 1119–1126.

[70] Artur d'Avila Garcez, Marco Gori, Luis C. Lamb, Luciano Serafini, Michael Spranger, and Son N. Tran. 2019. Neural-symbolic computing: An effective methodology for principled integration of machine learning and reasoning.

[71] Iván García-Magariño, Rajarajan Muttukrishnan, and Jaime Lloret. 2019. Human-centric AI for trustworthy IoT systems with explainable multilayer perceptrons. *IEEE Access* 7 (2019), 125562–125574.

[72] Michael Georgeff, Barney Pell, Martha Pollack, Milind Tambe, and Michael Wooldridge. 1998. The belief-desire-intention model of agency. In *International Workshop on Agent Theories, Architectures, and Languages*. Springer, 10 pages.

[73] Melinda T. Gervasio, Karen L. Myers, Eric Yeh, and Boone Adkins. 2018. Explanation to avert surprise. In *IUI Workshops*, Vol. 2068.

[74] Shalini Ghosh, Giedrius Burachas, Arijit Ray, and Avi Ziskind. 2019. Generating natural language explanations for visual question answering using scene graphs and visual attention.

[75] C. Lee Giles, Clifford B. Miller, Dong Chen, Hsing-Hen Chen, Guo-Zheng Sun, and Yee-Chun Lee. 1992. Learning and extracting finite state automata with second-order recurrent neural networks. *Neural Computation* 4, 3 (1992), 393–405.

[76] Michele Ginesi, Daniele Meli, Andrea Roberti, Nicola Sansonetto, and Paolo Fiorini. 2020. Autonomous task planning and situation awareness in robotic surgery. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 3144–3150.

[77] Manuel Giuliani, Mary Ellen Foster, Amy Isard, Colin Matheson, Jon Oberlander, and Alois Knoll. 2010. Situated reference in a hybrid human-robot interaction system. In *Proceedings of the 6th International Natural Language Generation Conference*. Association for Computational Linguistics, 67–75.

[78] Ze Gong and Yu Zhang. 2018. Behavior explanation as intention signaling in human-robot teaming. In *27th IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 1005–1011.

[79] Antoine Grea, Laëtitia Matignon, and Samir Aknine. 2018. How explainable plans can make planning faster. In *Workshop on Explainable Artificial Intelligence*. hal.archives-ouvertes.fr, Stockholm, Sweden, 58–64.

[80] Edward Groshev, Aviv Tamar, Maxwell Goldstein, Siddharth Srivastava, and Pieter Abbeel. 2018. Learning generalized reactive policies using deep neural networks. *Proceedings of the International Conference on Automated Planning and Scheduling* (2018), 408–416.

[81] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. 2018. A survey of methods for explaining black box models. *Comput. Surveys* 51, 5 (2018), 42 pages.

[82] Matthew Guzdial, Joshua Reno, Jonathan Chen, Gillian Smith, and Mark Riedl. 2018. Explainable PCGML Via Game Design Patterns.

[83] Maaike Harbers, Joost Broekens, Karel Van Den Bosch, and John-Jules Meyer. 2010. Guidelines for developing explainable cognitive models. In *Proceedings of the 10th International Conference on Composite Materials*. Citeseer, 85–90.

[84] Maaike Harbers, Karel Van Den Bosch, and John-Jules Meyer. 2009. A methodology for developing self-explaining agents for virtual training. In *International Workshop on Languages, Methodologies and Development Tools for Multi-Agent Systems*. Springer, 168–182.

[85] Maaike Harbers, Karel van den Bosch, and John-Jules Meyer. 2010. Design and evaluation of explainable BDI agents. In *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, Vol. 2. IEEE, 125–132.

[86] Peter Hase and Mohit Bansal. 2020. Evaluating explainable AI: Which algorithmic explanations help users predict model behavior?

[87] Helen Hastie, Francisco J. Chiyah Garcia, David A. Robb, Atanas Laskov, and Pedro Patron. 2018. MIRIAM: A multimodal interface for explaining the reasoning behind actions of remote autonomous systems. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*. 557–558.

[88] Bradley Hayes and Julie A. Shah. 2017. Improving robot controller transparency through autonomous policy explanation. In *12th ACM/IEEE International Conference on Human-Robot Interaction*. IEEE, 303–312.

[89] Steven R. Haynes, Mark A. Cohen, and Frank E. Ritter. 2009. Designs for explaining intelligent agents. *International Journal of Human-Computer Studies* 67, 1 (2009), 90–110.

[90] Alex L. G. Hayzelden and John Bigham. 1999. *Software Agents for Future Communication Systems*. Springer Science & Business Media.

[91] Aroua Hedhili, Wided Lejouad Chaari, and Khaled Ghédira. 2013. Explanation language syntax for multi-agent systems. In *2013 World Congress on Computer and Information Technology*. IEEE, 6 pages.

[92] Joshua Ho and Chien-Min Wang. 2020. Explainable and adaptable augmentation in knowledge attention network for multi-agent deep reinforcement learning systems. In *2020 IEEE Third International Conference on Artificial Intelligence and Knowledge Engineering*. IEEE, 157–161.

[93] Markus Hofmarcher, Thomas Unterthiner, José Arjona-Medina, Günter Klambauer, Sepp Hochreiter, and Bernhard Nessler. 2019. Visual scene understanding for autonomous driving using semantic segmentation. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Springer, 285–296.

[94] Daniel Holliday, Stephanie Wilson, and Simone Stumpf. 2013. The effect of explanations on perceived control and behaviors in intelligent systems. In *2013 Extended Abstracts on Human Factors in Computing Systems*. 181–186.

[95] Andreas Holzinger, Bernd Malle, Peter Kieseberg, Peter M. Roth, Heimo Müller, Robert Reihs, and Kurt Zatloukal. 2017. Towards the augmented pathologist: Challenges of explainable-AI in digital pathology.

[96] Yujia Huo, Derek F. Wong, Lionel M. Ni, Lidia S. Chao, and Jing Zhang. 2020. HeTROPY: Explainable learning diagnostics via heterogeneous maximum-entropy and multi-spatial knowledge representation. *Knowledge-Based Systems* 207 (2020), 106389.

[97] Nitin Indurkhya and Fred J. Damerau. 2010. *Handbook of Natural Language Processing*. Chapman and Hall/CRC.

[98] Ulit Jaidee, Héctor Muñoz-Avila, and David W. Aha. 2011. Case-based learning in goal-driven autonomy agents for real-time strategy combat tasks. In *Proceedings of the 19th International Conference on Case-Based Reasoning (ICCBR 2011) Workshop on Computer Games*. 43–52.

[99] Ulit Jaidee, Héctor Muñoz-Avila, and David W. Aha. 2011. Integrated learning for goal-driven autonomy. In *22nd International Joint Conference on Artificial Intelligence*.

[100] Adrien Jauffret, Nicolas Cuperlier, Philippe Gaussier, and Philippe Tarroux. 2013. From self-assessment to frustration, a small step toward autonomy in robotic navigation. *Frontiers In Neurorobotics* 7 (2013), 16.

[101] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. 2015. Image retrieval using scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3668–3678.

[102] W. Lewis Johnson. 1994. Agents that learn to explain themselves. In *Proceedings of the 12th AAAI National Conference on Artificial Intelligence*. 1257–1263.

[103] Zoe Juozapaitis, Anurag Koul, Alan Fern, Martin Erwig, and Finale Doshi-Velez. 2019. Explainable reinforcement learning via reward decomposition. In *Proceedings of the IJCAI 2019 Workshop on Explainable Artificial Intelligence*. 47–53.

[104] Subbarao Kambhampati and Smadar Kedar. 1994. A unified framework for explanation-based generalization of partially ordered and partially instantiated plans. *Artificial Intelligence* 67, 1 (1994), 29–70.

[105] Frank Kaptein, Joost Broekens, Koen Hindriks, and Mark Neerincx. 2017. Personalised self-explanation by robots: The role of goals versus beliefs in robot-action explanation for children and adults. In *26th IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 676–682.

[106] Frank Kaptein, Joost Broekens, Koen Hindriks, and Mark Neerincx. 2017. The role of emotion in self-explanations by cognitive agents. In *7th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos*. IEEE, 88–93.

[107] Frank C. Keil. 2006. Explanation and understanding. *Annual Review of Psychology* 57 (2006), 227–254.

[108] Catriona M. Kennedy. 1998. A conceptual foundation for autonomous learning in unforeseen situations. In *Proceedings of the IEEE International Symposium on Intelligent Control*. IEEE, 483–488.

[109] Matthias Kerzel, Jakob Ambsdorf, Dennis Becker, Wenhao Lu, Erik Strahl, Josua Spisak, Connor Gäde, Tom Weber, and Stefan Wermter. 2022. What's on your mind, NICO? *KI-Künstliche Intelligenz* (2022), 18 pages.

[110] Boaz Keysar, Shuhong Lin, and Dale J. Barr. 2003. Limits on theory of mind use in adults. *Cognition* 89, 1 (2003), 25–41.

[111] Taemie Kim and Pamela Hinds. 2006. Who should I blame? Effects of autonomy and transparency on attributions in human-robot interaction. In *Proceedings of the 15th IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 80–85.

[112] Matthew Klenk, Matt Molineaux, and David W. Aha. 2013. Goal-driven autonomy for responding to unexpected events in strategy simulations. *Computational Intelligence* 29, 2 (2013), 187–206.

[113] Kevin Knight. 1993. Are many reactive agents better than a few deliberative ones?. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, Vol. 93. 432–437.

[114] T. Kobayashi. 1985. Speech conversation system of the musician robot. *Proceedings of the 1985 International Conference on Advanced Robotics* (1985), 483–488.

[115] Teuvo Kohonen. 1990. The self-organizing map. *Proc. IEEE* 78, 9 (1990), 1464–1480.

[116] Janet Kolodner. 2014. *Case-Based Reasoning*. Morgan Kaufmann Publishers Inc.

[117] Avinash Kori, Parth Natekar, Ganapathy Krishnamurthi, and Balaji Srinivasan. 2020. Abstracting Deep Neural Networks into Concept Graphs for Concept Level Interpretability.

[118] Raj Korpan and Susan L. Epstein. 2018. Toward natural explanations for a robot's navigation plans. *Notes from the Explainable Robotic Systems Workshop, Human-Robot Interaction* (2018).

[119] Emiel Krahmer and Kees Van Deemter. 2012. Computational generation of referring expressions: A survey. *Computational Linguistics* 38, 1 (2012), 173–218.

[120] Alfred Kranstedt and Ipke Wachsmuth. 2005. Incremental generation of multimodal deixis referring to objects. In *Proceedings of the 10th European Workshop on Natural Language Generation*.

[121] Jens Kröske, Kevin O'Holleran, and Hannu Rajaniemi. 2009. Trusted reasoning engine for autonomous systems with an interactive demonstrator. In *4th SEAS DTC Technical Conference*. Citeseer.

[122] Todd Kulesza, Simone Stumpf, Margaret Burnett, Weng-Keen Wong, Yann Riche, Travis Moore, Ian Oberst, Amber Shinsel, and Kevin McIntosh. 2010. Explanatory debugging: Supporting end-user debugging of machine-learned programs. In *2010 IEEE Symposium on Visual Languages and Human-centric Computing*. IEEE, 41–48.

[123] Todd Kulesza, Simone Stumpf, Margaret Burnett, Sherry Yang, Irwin Kwan, and Weng-Keen Wong. 2013. Too much, too little, or just right? Ways explanations impact end users' mental models. In *2013 IEEE Symposium on Visual Languages and Human-centric Computing*. IEEE, 3–10.

[124] Brenden M. Lake, Tomer D. Ullman, Joshua B. Tenenbaum, and Samuel J. Gershman. 2017. Building machines that learn and think like people. *Behavioral and Brain Sciences* 40 (2017).

[125] Pat Langley, Ben Meadows, Mohan Sridharan, and Dongkyu Choi. 2017. Explainable agency for intelligent autonomous systems. In *Proceedings of the 29th Innovative Applications of Artificial Intelligence Conference*.

[126] Simon Meyer Lauritsen, Mads Kristensen, Mathias Vassard Olsen, Morten Skaarup Larsen, Katrine Meyer Lauritsen, Marianne Johansson Jørgensen, Jeppe Lange, and Bo Thiesson. 2020. Explainable artificial intelligence model to predict acute critical illness from electronic health records. *Nature Communications* 11, 1 (2020), 11 pages.

[127] Nada Lavrac and Saso Dzeroski. 1994. Inductive logic programming. In *Proceedings of the 10th Logic Programming Workshop*. Springer, 146–160.

[128] Jaesong Lee, Joong-Hwi Shin, and Jun-Seok Kim. 2017. Interactive visualization and manipulation of attention-based neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. 121–126.

[129] Iolanda Leite, Carlos Martinho, and Ana Paiva. 2013. Social robots for long-term interaction: A survey. *International Journal of Social Robotics* 5, 2 (2013), 291–308.

[130] Benjamin Lettl and Axel Schulte. 2013. Self-explanation capability for cognitive agents on-board of UCAVs to improve cooperation in a manned-unmanned fighter team. In *AIAA Infotech@ Aerospace (I@A) Conference*. 4898.

[131] Sirui Li, Weixing Sun, and Tim Miller. 2015. Communication in human-agent teams for tasks with joint action. In *International Workshop on Coordination, Organizations, Institutions, and Norms in Agent Systems*. Springer, 224–241.

[132] Yanfen Li, Hanxiang Wang, L. Minh Dang, Tan N. Nguyen, Dongil Han, Ahyun Lee, Insung Jang, and Hyeonjoon Moon. 2020. A deep learning-based hybrid framework for object detection and recognition in autonomous driving. *IEEE Access* 8 (2020), 194228–194239.

[133] Wei Shiung Liew, Chu Kiong Loo, and Stefan Wermter. 2021. Emotion recognition using explainable genetically optimized fuzzy ART ensembles. *IEEE Access* 9 (2021), 61513–61531.

[134] Brian Y. Lim and Anind K. Dey. 2011. Design of an intelligible mobile context-aware application. In *Proceedings of the 13th International Conference on Human Computer Interaction With Mobile Devices and Services*. 157–166.

[135] Long-Ji Lin. 1992. Self-improving reactive agents based on reinforcement learning, planning and teaching. *Machine Learning* 8, 3–4 (1992), 293–321.

[136] Zachary C. Lipton, David C. Kale, Charles Elkan, and Randall Wetzel. 2015. Learning to diagnose with LSTM recurrent neural networks.

[137] Michael L. Littman. 1994. Memoryless policies: Theoretical limitations and practical results. In *From Animals to Animats 3: Proceedings of the Third International Conference on Simulation of Adaptive Behavior*, Vol. 3. Cambridge, MA, USA, 238.

[138] Shusen Liu, Tao Li, Zhimin Li, Vivek Srikumar, Valerio Pascucci, and Peer-Timo Bremer. 2018. *Visual Interrogation of Attention-based Models for Natural Language Inference and Machine Comprehension*. Technical Report. Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States).

[139] Meghann Lomas, Robert Chevalier, Ernest Vincent Cross, Robert Christopher Garrett, John Hoare, and Michael Kopack. 2012. Explaining robot actions. In *Proceedings of the 7th Annual ACM/IEEE International Conference on Human-Robot Interaction*. 187–188.

[140] Patrick Lucey, Jeffrey F. Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews. 2010. The extended Cohn-Kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*. IEEE Computer Society, 94–101.

[141] Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*. 4765–4774.

[142] Prashan Madumal, Tim Miller, Liz Sonenberg, and Frank Vetere. 2019. Explainable reinforcement learning through a causal lens.

[143] Bertram F. Malle. 1999. How people explain behavior: A new theoretical framework. *Personality and Social Psychology Review* 3, 1 (1999), 23–48.

[144] James Malone, Ken McGarry, Stefan Wermter, and Chris Bowerman. 2006. Data mining using rule extraction from Kohonen self-organising maps. *Neural Computing & Applications* 15, 1 (Mar. 2006), 9–17. https://doi.org/10.1007/s00521-005-0002-1

[145] Jiayuan Mao, Yuan Yao, Stefan Heinrich, Tobias Hinz, Cornelius Weber, Stefan Wermter, Zhiyuan Liu, and Maosong Sun. 2019. Bootstrapping knowledge graphs from images and text. *Frontiers in Neurorobotics* 13, 93 (Nov. 2019). https://doi.org/10.3389/fnbot.2019.00093

[146] Ken McGarry, Stefan Wermter, and Sheila Garfield. 2007. Auto extraction, representation and integration of a diabetes ontology using Bayesian networks. In *Proceedings of the IEEE International Symposium on Computer Based Medical Systems*. IEEE, Maribor, Slovenia, 612–617. https://doi.org/10.1109/CBMS.2007.26

[147] Ken McGarry, Stefan Wermter, and J. MacIntyre. 2001. Knowledge extraction from local function networks. In *Proceedings of the International Joint Conference on Artificial Intelligence*. Seattle, WA, USA. https://www2.informatik.uni-hamburg.de/wtm/publications/2001/MWM01a/.

[148] Masahiko Mikawa, Yuriko Yoshikawa, and Makoto Fujisawa. 2018. Expression of intention by rotational head movements for teleoperated mobile robot. In *IEEE 15th International Workshop on Advanced Motion Control*. IEEE, 249–254.

[149] Nykan Mirchi, Vincent Bissonnette, Recai Yilmaz, Nicole Ledwos, Alexander Winkler-Schwartz, and Rolando F. Del Maestro. 2020. The virtual operative assistant: An explainable artificial intelligence tool for simulation-based training in surgery and medicine. *PLOS One* 15, 2 (2020), e0229596.

[150] Brent Daniel Mittelstadt and Luciano Floridi. 2017. Transparent, explainable, and accountable AI for robotics. *Science Robotics* 2, 6 (2017).

[151] Matthew Molineaux, Dustin Dannenhauer, and David W. Aha. 2018. Towards explainable NPCs: A relational exploration learning agent. In *Workshops at the 32nd AAAI Conference on Artificial Intelligence*. 565–569.

[152] Matthew Molineaux, Matthew Klenk, and David Aha. 2010. Goal-driven autonomy in a navy strategy simulation. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*. AAAI Press, 1548–1554.

[153] Raymond J. Mooney and Scott Bennett. 1986. A domain independent explanation-based generalizer. In *Proceedings of the 5th AAAI Conference on Artificial Intelligence*. 551–555.

[154] Hector Munoz-Avila and David W. Aha. 2010. A case study of goal-driven autonomy in domination games. In *Proceedings of the AAAI Workshop on Goal-Directed Autonomy*.

[155] Dana Nau, Yue Cao, Amnon Lotem, and Hector Munoz-Avila. 1999. SHOP: Simple hierarchical ordered planner. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence*. 968–973.

[156] Dana S. Nau. 2007. Current trends in automated planning. *AI Magazine* 28, 4 (2007), 43–43.

[157] Mark A. Neerincx, Jasper van der Waa, Frank Kaptein, and Jurriaan van Diggelen. 2018. Using perceptual and cognitive explanations for enhanced human-agent team performance. In *International Conference on Engineering Psychology and Cognitive Ergonomics*. Springer, 204–214.

[158] Hwei Geok Ng, Matthias Kerzel, Jan Mehnert, Arne May, and Stefan Wermter. 2018. Classification of MRI migraine medical data using 3D convolutional neural network. In *International Conference on Artificial Neural Networks*. Springer, 300–309.

[159] Mehrbakhsh Nilashi, Dietmar Jannach, Othman bin Ibrahim, Mohammad Dalvi Esfahani, and Hossein Ahmadi. 2016. Recommendation quality, transparency, and website quality for trust-building in recommendation agents. *Electronic Commerce Research and Applications* 19 (2016), 70–84.

[160] Ryuichi Nisimura, Takashi Uchida, Akinobu Lee, Hiroshi Saruwatari, Kiyohiro Shikano, and Yoshio Matsumoto. 2002. ASKA: Receptionist robot with speech dialogue system. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 2. IEEE, 1314–1319.

[161] Jekaterina Novikova, Leon Watts, and Tetsunari Inamura. 2015. Emotionally expressive robot behavior improves human-robot collaboration. In *24th IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 7–12.

[162] Christian W. Omlin and C. Lee Giles. 1996. Extraction of rules from discrete-time recurrent neural networks. *Neural Networks* 9, 1 (1996), 41–52.

[163] Junichi Osada, Shinichi Ohnaka, and Miki Sato. 2006. The scenario and design process of childcare robot, PaPeRo. In *Proceedings of the 2006 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*. 80.

[164] Mayada Oudah, Talal Rahwan, Tawna Crandall, and Jacob W. Crandall. 2018. How AI wins friends and influences people in repeated games with cheap talk. In *32nd AAAI Conference on Artificial Intelligence*. AAAI Press, 1519–1526.

[165] Xinlei Pan, Tingnan Zhang, Brian Ichter, Aleksandra Faust, Jie Tan, and Sehoon Ha. 2020. Zero-shot imitation learning from demonstrations for legged robot visual navigation. In *2020 IEEE International Conference on Robotics and Automation*. IEEE, 679–685.

[166] Rey Pocius, Lawrence Neal, and Alan Fern. 2019. Strategic tasks for explainable reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 10007–10008.

[167] Alun Preece, Dan Harborne, Dave Braines, Richard Tomsett, and Supriyo Chakraborty. 2018. Stakeholders in explainable AI.

[168] Frank Puppe. 2012. *Systematic Introduction To Expert Systems: Knowledge Representations and Problem-Solving Methods*. Springer Science & Business Media.

[169] David V. Pynadath, Ning Wang, Ericka Rovira, and Michael J. Barnes. 2018. Clustering behavior to recognize subjective beliefs in human-agent teams. In *Proceedings of the 17th International Conference on Autonomous Agents and Multi-Agent Systems*. 1495–1503.

[170] Lilin Qian, Xin Xu, Yujun Zeng, and Junwen Huang. 2019. Deep, consistent behavioral decision making with planning features for autonomous vehicles. *Electronics* 8, 12 (2019), 1492.

[171] Lara Quijano-Sanchez, Christian Sauer, Juan A. Recio-Garcia, and Belen Diaz-Agudo. 2017. Make it personal: A social explanation system applied to group recommendations. *Expert Systems with Applications* 76 (2017), 36–48.

[172] Anand S. Rao, Michael P. Georgeff, et al. 1995. BDI agents: From theory to practice. In *Proceedings of the 1st International Conference on Multi-Agent Systems*. 312–319.

[173] Kasim Rehman, Frank Stajano, and George Coulouris. 2005. Visually interactive location-aware computing. In *International Conference on Ubiquitous Computing*. Springer, 177–194.

[174] Raymond Reiter and Johan De Kleer. 1986. An assumption-based truth-maintenance system. *Artificial Intelligence* (1986), 127–162.

[175] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2016. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*. 91–99.

[176] Marcel Jurriaan Robeer. 2018. *Contrastive Explanation for Machine Learning*. Master's thesis. Utrecht University, Netherlands.

[177] Stephanie Rosenthal, Sai P. Selvaraj, and Manuela M. Veloso. 2016. Verbalization: Narration of autonomous robot experience. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence*. 862–868.

[178] Alessandra Maria Sabelli, Takayuki Kanda, and Norihiro Hagita. 2011. A conversational robot in an elderly care center: An ethnographic study. In *6th ACM/IEEE International Conference on Human-Robot Interaction*. IEEE, 37–44.

[179] Michael K. Sahota. 1994. Reactive deliberation: An architecture for real-time intelligent control in dynamic environments. In *Proceedings of the 12th National Conference on Artificial Intelligence*, Vol. 2. 1303–1308.

[180] Yoshiaki Sakagami, Ryujin Watanabe, Chiaki Aoyama, Shinichi Matsunaga, Nobuo Higaki, and Kikuo Fujimura. 2002. The intelligent ASIMO: System overview and integration. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 3. IEEE, 2478–2483.

[181] Vieri Giuliano Santucci, Gianluca Baldassarre, and Marco Mirolli. 2016. GRAIL: A goal-discovering robotic architecture for intrinsically-motivated learning. *IEEE Transactions on Cognitive and Developmental Systems* 8, 3 (2016), 214–231.

[182] Miki Sato, Akihiko Sugiyama, and Shin'ichi Ohnaka. 2006. Auditory system in a personal robot, PaPeRo. In *2006 Digest of Technical Papers International Conference on Consumer Electronics*. IEEE, 19–20.

[183] Aroua Hedhili Sbai, Wided Lejouad Chaari, and Khaled Ghedira. 2013. Intra-agent explanation using temporal and extended causal maps. *Procedia Computer Science* 22 (2013), 241–249.

[184] Jean Scholtz. 2003. Theory and evaluation of human robot interactions. In *Proceedings of the 36th Annual Hawaii International Conference on System Sciences*. IEEE, 10.

[185] Bastian Seegebarth, Felix Müller, Bernd Schattenberg, and Susanne Biundo. 2012. Making hybrid plans more clear to human users - A formal approach for generating sound explanations. In *Proceedings of the 22nd International Conference on International Conference on Automated Planning and Scheduling*. 225–233.

[186] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*. 618–626.

[187] Pedro Sequeira and Melinda Gervasio. 2019. Interestingness Elements for Explainable Reinforcement Learning: Understanding Agents' Capabilities and Limitations.

[188] Ying Sha and May D. Wang. 2017. Interpretable predictions of clinical outcomes with an attention-based recurrent neural network. In *Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*. 233–240.

[189] Ivan Shindev, Yu Sun, Michael Coovert, Jenny Pavlova, and Tiffany Lee. 2012. Exploration of intention expression for robots. In *Proceedings of the 7th Annual ACM/IEEE International Conference on Human-Robot Interaction.* 247–248.

[190] Katsuhiko Shirai and Hiromichi Fujisawa. 1974. An algorithm for spoken sentence recognition and its application to the speech input-output system. *IEEE Transactions on Systems, Man, & Cybernetics* 4, 5 (1974), 475–479.

[191] Mohit Shridhar, Dixant Mittal, and David Hsu. 2020. INGRESS: Interactive visual grounding of referring expressions. *The International Journal of Robotics Research* 39, 2–3 (2020), 217–232.

[192] Henrique Siqueira, Sven Magg, and Stefan Wermter. 2020. Efficient facial feature learning with wide ensemble-based convolutional neural networks.

[193] Shirin Sohrabi, Jorge A. Baier, and Sheila A. McIlraith. 2011. Preferred explanations: Theory and generation via planning. In *25th AAAI Conference on Artificial Intelligence.*

[194] Sichao Song and Seiji Yamada. 2018. Effect of expressive lights on human perception and interpretation of functional robot. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems.* 6 pages.

[195] Francesco Sovrano, Alex Raymond, and Amanda Prorok. 2021. Explanation-Aware Experience Replay in Rule-Dense Environments.

[196] Mohan Sridharan and Ben Meadows. 2019. Towards a theory of explanations for human-robot collaboration. *KI-Künstliche Intelligenz* 33, 4 (2019), 331–342.

[197] Simone Stumpf, Weng-Keen Wong, Margaret Burnett, and Todd Kulesza. 2010. Making intelligent systems understandable and controllable by end users. https://web.engr.oregonstate.edu/~wongwe/papers/pdf/pervasive2012.pdf.

[198] Roykrong Sukkerd, Reid Simmons, and David Garlan. 2018. Toward explainable multi-objective probabilistic planning. In *2018 IEEE/ACM 4th International Workshop on Software Engineering for Smart Cyber-Physical Systems.* IEEE, 19–25.

[199] Alexander Sutherland, Suna Bensch, Thomas Hellström, Sven Magg, and Stefan Wermter. 2020. Tell me why you feel that way: Processing compositional dependency for tree-LSTM aspect sentiment triplet extraction (TASTE). In *Artificial Neural Networks and Machine Learning - ICANN 2020.* 660–671. https://doi.org/10.1007/978-3-030-61609-0_52

[200] William R. Swartout and Johanna D. Moore. 1993. Explanation in second generation expert systems. In *Second Generation Expert Systems.* Springer, 543–585.

[201] Aaquib Tabrez and Bradley Hayes. 2019. Improving human-robot interaction through explainable reinforcement learning. In *14th ACM/IEEE International Conference on Human-Robot Interaction.* IEEE, 751–753.

[202] Leila Takayama, Doug Dooley, and Wendy Ju. 2011. Expressing thought: Improving robot readability with animation principles. In *Proceedings of the 6th International Conference on Human-Robot Interaction.* 69–76.

[203] Massimo Tipaldi, Lorenzo Feruglio, Pierre Denis, and Gianni D'Angelo. 2020. On applying AI-driven flight data analysis for operational spacecraft model-based diagnostics. *Annual Reviews in Control* (2020).

[204] Joe Townsend, Thomas Chaton, and João M. Monteiro. 2019. Extracting relational explanations from deep neural networks: A survey from a neural-symbolic perspective. *IEEE Transactions on Neural Networks and Learning Systems* 31, 9 (2019), 3456–3470.

[205] Cristina Urdiales, Eduardo J. Perez, Javier Vázquez-Salceda, Miquel Sànchez-Marrè, and Francisco Sandoval. 2006. A purely reactive navigation scheme for dynamic environments using case-based reasoning. *Autonomous Robots* 21, 1 (2006), 65–78.

[206] Rik Van den Brule, Gijsbert Bijlstra, Ron Dotsch, Daniël H. J. Wigboldus, and W. F. G. Haselager. 2013. Signaling robot trustworthiness: Effects of behavioral cues as warnings. *Lecture Notes in Computer Science* 8239 (2013), 583–584.

[207] Jasper van der Waa, Jurriaan van Diggelen, Mark A. Neerincx, and Stephan Raaijmakers. 2018. ICM: An intuitive model independent and accurate certainty measure for machine learning. In *10th International Conference on Agents and Artificial Intelligence.* 314–321.

[208] Michael Van Lent, William Fisher, and Michael Mancuso. 2004. An explainable artificial intelligence system for small-unit tactical behavior. In *Proceedings of the National Conference on Artificial Intelligence.* Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 900–907.

[209] Michael van Lent and John Laird. 1998. Learning by observation in a complex domain. In *Proceedings of the Knowledge Acquisition Workshop.*

[210] Jo Vermeulen. 2010. Improving intelligibility and control in UbiComp. In *Proceedings of the 12th ACM International Conference Adjunct Papers on Ubiquitous Computing-Adjunct.* 485–488.

[211] Dirk Voelz, Elisabeth André, Gerd Herzog, and Thomas Rist. 1998. Rocco: A RoboCup soccer commentator system. In *Robot Soccer World Cup.* Springer, 50–60.

[212] Ning Wang, David V. Pynadath, and Susan G. Hill. 2016. The impact of POMDP-generated explanations on trust and performance in human-robot teams. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multi-Agent Systems.* 997–1005.

[213] Yang Wang, David Mulvaney, Ian Sillitoe, and Erick Swere. 2008. Robot navigation by waypoints. *Journal of Intelligent and Robotic Systems* 52, 2 (2008), 175–207.

[214] Ben George Weber, Michael Mateas, and Arnav Jhala. 2012. Learning from demonstration for goal-driven autonomy. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence.* AAAI Press, 1176–1182.

[215] Gail Weiss, Yoav Goldberg, and Eran Yahav. 2018. Extracting automata from recurrent neural networks using queries and counterexamples. In *International Conference on Machine Learning.* 5247–5256.

[216] Stefan Wermter. 1999. Preference Moore machines for neural fuzzy integration. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence.* 840–845.

[217] Stefan Wermter. 2000. Knowledge extraction from transducer neural networks. *Applied Intelligence* 12, 1–2 (2000), 27–42.

[218] Stefan Wermter, Sascha Griffiths, and Stefan Heinrich. 2017. Crossmodal lifelong learning in hybrid neural embodied architectures. *Behavioral and Brain Sciences* 40 (2017), 72.

[219] Handy Wicaksono, Claude Sammut, and Raymond Sheh. 2017. Towards explainable tool creation by a robot. In *International Joint Conference on Artificial Intelligence Workshop on Explainable AI.* 63.

[220] Ryan W. Wohleber, Kimberly Stowers, Jessie Y. C. Chen, and Michael Barnes. 2017. Effects of agent transparency and communication framing on human-agent teaming. In *2017 IEEE International Conference on Systems, Man, and Cybernetics.* IEEE, 3427–3432.

[221] M. Wooldridge. 1995. Conceptualising and developing agents. In *Proceedings of the UNICOM Seminar on Agent Software*, Vol. 42. London.

[222] Robert H. Wortham, Andreas Theodorou, and Joanna J. Bryson. 2016. What does the robot think? Transparency as a fundamental design requirement for intelligent systems. In *IJCAI Ethics for Artificial Intelligence Workshop.*

[223] Robert H. Wortham, Andreas Theodorou, and Joanna J. Bryson. 2017. Improving robot transparency: Real-time visualisation of robot AI substantially improves understanding in naive observers. In *26th IEEE International Symposium on Robot and Human Interactive Communication.* IEEE, 1424–1431.

[224] Danfei Xu, Yuke Zhu, Christopher B. Choy, and Li Fei-Fei. 2017. Scene graph generation by iterative message passing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 5410–5419.

[225] Jun Xu, Haifeng Wang, Zhengyu Niu, Hua Wu, and Wanxiang Che. 2020. Knowledge graph grounded goal planning for open-domain conversation generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 9338–9345.

[226] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. 2018. Graph R-CNN for scene graph generation. In *Proceedings of the European Conference on Computer Vision.* 670–685.

[227] Levent Yilmaz and Sunit Sivaraj. 2019. A cognitive architecture for verifiable system ethics via explainable autonomy. In *2019 IEEE International Systems Conference (SysCon).* IEEE, 8 pages.

[228] Licheng Yu, Hao Tan, Mohit Bansal, and Tamara L. Berg. 2017. A joint speaker-listener-reinforcer model for referring expressions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 7282–7290.

[229] Zheng Zeng, Rodney M. Goodman, and Padhraic Smyth. 1993. Learning finite state machines with self-clustering recurrent networks. *Neural Computation* 5, 6 (1993), 976–990.

[230] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. 2016. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2921–2929.

[231] Luowei Zhou, Nathan Louis, and Jason J. Corso. 2018. Weakly-supervised video object grounding from text by loss weighting and object interaction.