Theory-based Causal Transfer:
Integrating Instance-level Induction and Abstract-level Structure Learning

Mark Edmonds,[1,2] **Xiaojian Ma,**[1] **Siyuan Qi,**[1,2] **Yixin Zhu,**[1,2] **Hongjing Lu,**[3] **Song-Chun Zhu**[1,2]

[1]UCLA Center for Vision, Cognition, Learning, and Autonomy
[2]International Center for AI and Robot Autonomy (CARA)
[3]UCLA Computational Vision and Learning (CVL) Lab
{markedmonds,maxiaojian,syqi,yixin.zhu,hongjing}@ucla.edu, sczhu@stat.ucla.edu

Abstract

Learning transferable knowledge across similar but different settings is a fundamental component of generalized intelligence. In this paper, we approach the transfer learning challenge from a causal theory perspective. Our agent is endowed with two basic yet general theories for transfer learning: (i) a task shares a common abstract structure that is invariant across domains, and (ii) the behavior of specific features of the environment remain constant across domains. We adopt a Bayesian perspective of causal theory induction and use these theories to transfer knowledge between environments. Given these general theories, the goal is to train an agent by interactively exploring the problem space to (i) discover, form, and transfer useful *abstract and structural* knowledge, and (ii) induce useful knowledge from the *instance-level attributes* observed in the environment. A hierarchy of Bayesian structures is used to model abstract-level structural causal knowledge, and an instance-level associative learning scheme learns which specific objects can be used to induce state changes through interaction. This model-learning scheme is then integrated with a model-based planner to achieve a task in the Open-Lock environment, a virtual "escape room" with a complex hierarchy that requires agents to reason about an abstract, generalized causal structure. We compare performances against a set of predominate model-free reinforcement learning (RL) algorithms. RL agents showed poor ability transferring learned knowledge across different trials. Whereas the proposed model revealed similar performance trends as human learners, and more importantly, demonstrated transfer behavior across trials and learning situations.[1]

1 Introduction

The ability of agents to learn and *reuse* knowledge is a fundamental characteristic of general intelligence and is essential for agents to succeed in novel circumstances (Legg and Hutter 2007). Humans demonstrate a remarkable ability to transfer causal knowledge between environments governed by the same underlying mechanics, in spite of observational changes to the features of the environment (Edmonds et al. 2018). Early psychological research framed

[1]The proposed algorithm, all baseline algorithms, and supplementary can be found on the first author's website.
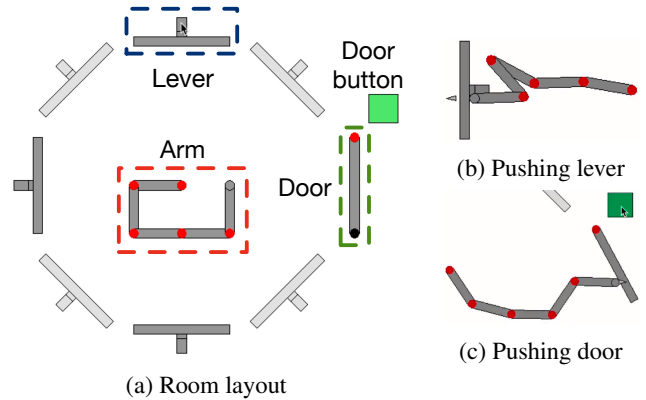

Figure 1: (a) Starting configuration of a 3-lever OpenLock room. The arm can interact with levers by either *pushing* outward or *pulling* inward, achieved by clicking either the outer or inner regions of the levers' radial tracks, respectively. Light gray levers are always locked; however, this is unknown to agents. The door can be pushed only after being unlocked. The green button serves as the mechanism to push on the door. The black circle on the door indicates whether or not the door is unlocked; locked if present, unlocked if absent. (b) Pushing on a lever. (c) Opening the door.

causal understanding as learning stimulus-response relationships through observation in classical conditioning experimental paradigms (Shanks and Dickinson 1988; Rescorla and Wagner 1972). However, more recent studies show human understanding of causal mechanisms in the distal world is more complex than covariation between observed (perceptual) variables (Holyoak and Cheng 2011); *e.g.*, humans *explore* and *experiment* with dynamic physical scenarios to refine causal hypotheses (Bramley et al. 2018; Stahl and Feigenson 2015).

Since the associative account, researchers have demonstrated that humans uncover causal relationships through the discovery of abstract causal structure (Waldmann and Holyoak 1992) and causal strength (Cheng 1997). Simultaneously, causal graphical models and Bayesian statistical inference have been developed to provide a general representational framework for how causal structure and strength are discovered (Griffiths and Tenenbaum 2005;

2009; Tenenbaum, Griffiths, and Kemp 2006; Bramley, Lagnado, and Speekenbrink 2015; Bramley et al. 2017; Holyoak and Cheng 2011). Under such a framework, causal connections encode a structural model of the world. States represent some status in the world, and connections between states imply the presence of a causal relationship. However, a critical component in causal learning is active *interaction* with the physical world, based on whether perceived information matches predictions from causal hypotheses. In this work, we combine causal learning (a form of model-building) with a model-based planner to effectively achieve tasks in environments where dynamics are unknown.

In contrast to this work beyond the associative account of causal understanding, recent success in the field of deep reinforcement learning (RL) has produced a wide body of research, showcasing agents learning how to play games (Mnih et al. 2015; Silver et al. 2016; Schulman et al. 2015; 2017) and develop complex robotic motor skills (Levine et al. 2016; Lillicrap et al. 2015) using associative learning schemes. However, the majority of model-free RL methods still have great difficulty transferring learned policies to new environments with consistent underlying mechanics but some dissimilar surface features (Zhang et al. 2018; Kansky et al. 2017). This deficiency is due to the limited scope of the agent's overall objective: learning which actions will likely lead to future rewards based on the current state of the environment. In traditional RL architectures, changes to the location and orientation of critical elements (instance-level) in the agent's environment *appear* as entirely new states, even though their functionality often remains the same (in the abstract-level). Since model-free RL agents do not attempt to encode transferable rules governing their environment, new situations appear as entirely new worlds. Although an agent can devise expert-level strategies through experiences in an environment, once that environment is perturbed, the agent must repeat an extensive learning process to relearn an effective policy in the altered environment.

In this work, the transfer learning problem is viewed as a combination of instance-level associative learning and abstract-level causal learning. We propose: (i) a bottom-up associative learning scheme that determines which attributes are associated with changes in the environment, and (ii) a top-down causal structure learning scheme that infers which atomic causal structures are useful for a task. The outcomes of actions are used to update beliefs about the causal hypothesis space, and our agent learns a dynamics model capable of solving our task. Specifically, we utilize a virtual "escape room" where agents are trapped in an empty room with a locked door. There is a series of conspicuous levers placed around the room with which an agent may interact. Agents placed in such a room may randomly push or pull on the levers to revise their theory about the door's locking mechanism based on observed changes in the environment's state. Once an agent discovers a solution, the agent is placed back into the same room but tasked with finding the *next* (different) solution. The agent "escapes" from the room after finding *all* of the solutions that can be used to unlock the door.

After completing (escaping) a single room, the agent is placed into a similar room, but with newly positioned levers.

Although the levers are in different positions, the rules governing this new room are the same as the last. Thus, the agent's task is to identify the role of each lever, according to the previously learned rules. Because these rules are abstract descriptions of the latent state of the escape room, we refer to the underlying theory as a causal schema (Heider 1958); *i.e.*, a conceptual organization of events identified as cause and effect. Once learned, an agent is able to transfer the learned schema despite different arrangements of levers in the room. Finally, we task agents with transferring knowledge with a different but similar causal schema. The new schema may add additional levers (nodes in a graphical model) or, in a more challenging way, rearrange the structure.

This paper integrates multiple modeling approaches to produce a highly capable agent that can learn causal schemas and transfer knowledge to new scenarios. The contribution of this paper is threefold:

1. Learning a bottom-up associative theory that encodes which objects and actions contribute to causal relations;
2. Learning which top-down atomic causal schemas are solutions, thereby learning generalized abstract task structure;
3. Integrating the top-down and bottom-up learning scheme with a model-based planner to optimally select interventions from causal hypotheses.

The remainder of this paper is organized as follows: Section 2 describes the OpenLock task. We present the proposed method of causal theory induction and intervention selection in Section 3 and Section 4, respectively. Section 5 compares the performance of the proposed model against various RL algorithms. Section 6 concludes the paper with discussions.

2 OpenLock Task

The OpenLock task, originally presented in Edmonds et al. 2018, requires agents to "escape" from a virtual room by unlocking and opening a door. The door is unlocked by manipulating the levers in a particular sequence (see Fig. 1a). Each lever can be manipulated using the robotic arm to *push* or *pull* on levers. Only a subset of the levers, specifically grey levers, are involved in unlocking the door (*i.e.*, active levers). White levers are never involved in unlocking the door (*i.e.*, inactive levers); however, this information is not provided to agents. Thus, at the instance-level, agents are expected to learn that grey levers are always part of solutions and white levers are not. Agents are also tasked with finding *all* solutions in the room, instead of a single solution.

Schemas: The door locking mechanism is governed by two causal schemas: Common Cause (CC) and Common Effect (CE). We use the terms Common Cause 3 (CC3) and Common Effect 3 (CE3) for schemas with three levers involved in solutions, and Common Cause 4 (CC4) and Common Effect 4 (CE4) with four levers; see Fig. 2. Three-lever trials have two solutions; four-lever trials have three solutions. Agents are required to find all solutions within a specific room to ensure that they form either CC or CE schema structure; a single solution corresponds to a causal chain.

Constraints: Agents also operate under an action-limit constraint, where only 3 actions (referred to as an *attempt*) can be used to (i) *push* or *pull* on (active or inactive) levers, or
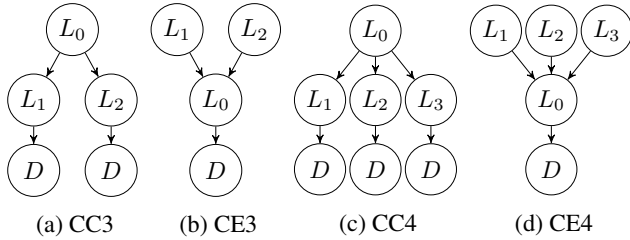
Figure 2: (a) Common Cause 3 (CC3) causal structure. (b) Common Effect 3 (CE3) causal structure. (c) Common Cause 4 (CC4) causal structure. (d) Common Effect 4 (CE4) causal structure. L_0, L_1, L_2 denote different locks, and D the door.

(ii) *push* open the door. This action-limit constraint prevents the search depth of interactions with the environment. After 3 actions, regardless of the outcome, the attempt terminates, and the environment resets. Regardless of whether the agent finds all solutions, agents are also constrained to a limited number of attempts in a particular room (referred to as a *trial*; *i.e.*, a sequence of attempts in a room, resulting in finding all the solutions or running out of attempts). An optimal agent will use at most $N + 1$ attempts to complete a trial, where N is the number of solutions in the trial. One attempt would be used to identify the role of every lever in the abstract schema, and N attempts would be used for each solution.

Training: Training sessions contain only 3-lever trials. After finishing a trial, the agent is placed in another trial (*i.e.*, room) with the *same* underlying causal schema but with a different arrangement of levers. If agents are forming a useful abstraction of task structure, the knowledge they acquired in previous trials should accelerate their ability to find all solutions in the present and future trials.

Transfer: In the transfer phase, we examine agents' ability to generalize the learned abstract causal schema to *different* but similar environments. We use four transfer conditions consisting of (i) congruent cases where the transfer schema adopts the same structure but with an additional lever (CE3-CE4 and CC3-CC4), and (ii) incongruent cases where the underlying schema is changed with an additional lever (CC3-CE4 and CE3-CC4). We compare these transfer results against two baseline conditions (CC4 and CE4), where the agent is trained in a sequence of 4-lever trials.

While seemingly simple, this task is unique and challenging for several reasons. First, requiring the agent to find all solutions rather than a single solution enforces the task as a CC or CE structure, instead of a single causal chain. Second, transferring the agent between trials with the same underlying causal schema but different lever positions encourages efficient agents to learn an *abstract* representation of the causal schema, rather than learning *instance-level* policies tailored to a specific trial. We would expect agents unable to form this abstraction to perform poorly in any transfer condition. Third, the congruent and incongruent transfer conditions test how well agents are able to adapt their learned knowledge to different but similar causal circumstances. These characteristics of the OpenLock task present challenges for current machine learning algorithms, especially model-free RL algorithms.

3 Causal Theory Induction

Causal theory induction provides a Bayesian account of how hierarchical causal theories can be induced from data (Griffiths and Tenenbaum 2005; 2009; Tenenbaum, Griffiths, and Kemp 2006). The key insight is: *hierarchy enables abstraction*. At the highest level, a theory provides general background knowledge about a task or environment. Theories consist of principles, principles lead to structure, and structure leads to data. The hierarchy used here is shown in Fig. 3a. Our agent utilizes two theories to learn a model of the Open-Lock environment: (i) an instance-level associative theory regarding which attributes and actions induce state changes in the environment, denoted as the bottom-up β theory, and (ii) an abstract-level causal structure theory about which atomic causal structures are useful for the task, denoted as the top-down γ theory.

Notation, Definition, and Space: A hypothesis space, Ω_C, is defined over possible causal chains, $c \in \Omega_C$. Each chain is defined as a tuple of subchains: $c = (c_0, \ldots, c_k)$, where k is the length of the chain, and each subchain is defined as a tuple $c_i = (a_i, s_i, cr_i^a, cr_i^s)$. Each a_i is an action node that the agent can execute, s_i is a state node, cr_i^a is a causal relation that defines how a state s_i transitions under an action a_i, and cr_i^s is a causal relation that defines how state s_i is affected by changes to the previous state, s_{i-1}. Each s_i is defined by a set of time-invariant *attributes*, ϕ_i and time-varying *fluents*, f_i (Thielscher 1998; Maclaurin 1742; Newton and Colson 1736); *i.e.*, $s_i = (\phi_i, f_i)$. Action nodes can be directly intervened on, but state nodes cannot. This means an agent can directly influence (*i.e.*, execute) an action, but how the action affects the world must be *actively* learned. The structure of the general causal chain is shown in the uninstantiated causal chain in Fig. 3a. As an example using Fig. 1a and the first causal chain in the causal chain level of Fig. 3a, if the agent executes *push* on the *upper* lever, the *lower* lever may transition from *pulled* to *pushed*, and the *left* lever may transition from *locked* to *unlocked*.

The space of states is defined as $\Omega_S = \Omega_\phi \times \Omega_F$, where the space of attributes Ω_ϕ consists of position and color, and the space of fluents Ω_F consists of binary values for lever status (*pushed* or *pulled*) and lever lock status (*locked* or *unlocked*). The space of causal relations is defined as $\Omega_{CR} = \Omega_F \times \Omega_F$, capturing the possibly binary transitions between previous fluent values and the next fluent values.

State nodes encapsulate both the time-invariant (attributes) and time-varying (fluents) components of an object. Attributes are defined by low-level features (*e.g.*, position, color, and orientation). These low-level attributes provide general background knowledge about how specific objects change under certain actions; *e.g.*, which levers can be pushed/pulled.

Method Overview: Our agent induces instance-level knowledge regarding which objects (*i.e.*, instances) can produce causal state changes through interaction (see Section 3.1) and simultaneously learns an abstract structural understanding of the task (*i.e.*, schemas; see Section 3.2). The two learning mechanisms are combined to form a causal theory of the environment, and the agent uses this theory to reason about the optimal action to select based on past experiences (*i.e.*, interventions; see Section 4). After taking an
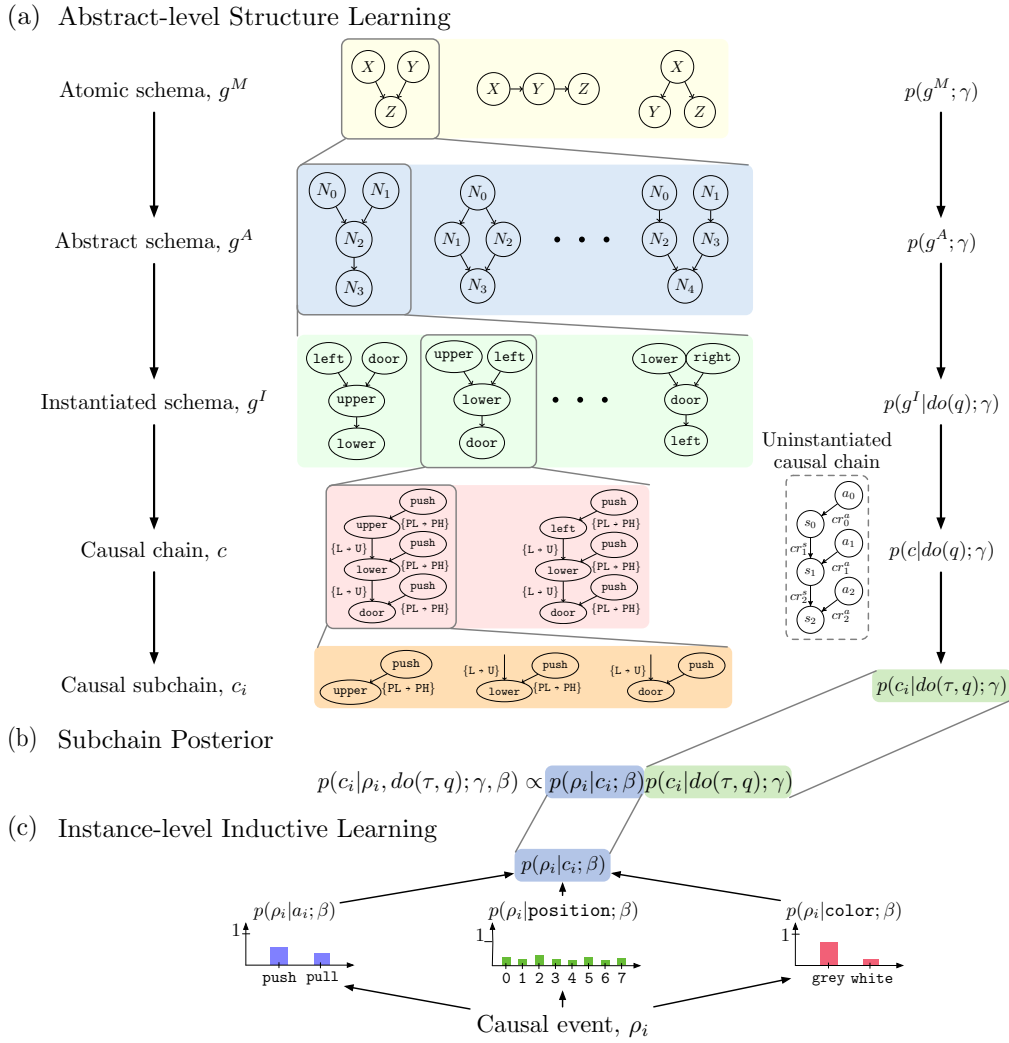
Figure 3: Illustration of top-down and bottom-up processes. (a) Abstract-level structure learning hierarchy. At the top, atomic schemas provide the agent with environment-invariant task structures. At the bottom, causal subchains represent a single time-step in the environment. The agent constructs the hierarchy and makes decisions at the causal subchain resolution. Atomic schemas g^M provide the top-level structural knowledge. Abstract schemas g^A are structures specific to a task, but not a particular environment. Instantiated schemas g^I are structures specific to a task and a particular environment. Causal chains c are structures representing a single attempt; an abstract, uninstantiated causal chain is also shown for notation. Each subchain c_i is a structure corresponding to a single action. PL, PH, L, U denote fluents *pulled*, *pushed*, *locked*, and *unlocked*, respectively. (b) The subchain posterior computed using the abstract-level structure learning and instance-level inductive learning. (c) Instance-level inductive learning. Each likelihood term is learned from causal events, ρ_i. Likelihood terms are combined for actions, positions, and colors.

action, the agent observes the effects and updates its model of both instance-level and abstract-level knowledge.

3.1 Instance-level Inductive Learning

The agent seeks to learn which instance-level components of the scene are associated with causal events; *i.e.*, we wish to learn a likelihood term to encode the probability that a causal event will occur. We adhere to a basic yet general associative learning theory: *causal relations induce state changes in the environment, and non-causal relations do not*, referred to as the bottom-up β theory. We learn two independent components: attributes and actions, and we assume they are independent to learn a general associative theory, rather than specific knowledge regarding an exact causal circumstance.

We define Ω_ϕ, the space of attributes, such as position and color, and learn which attributes are associated with levers that induce state changes in the environment. Specifically, an object is defined by its observable features; *i.e.*, the attributes ϕ. We also define Ω_A, a set of actions and learn a background likelihood over which actions are more likely to induce a state change. We assume attributes and actions are independent and learn each independently.

Our agent learns a likelihood term for each attribute ϕ_{ij} and action a_i using Dirichlet distributions because they serve as a conjugate prior to the multinomial distribution. First, a global Dirichlet parameterized by α^G is used across all trials to encode long-term beliefs about various environments. Upon entering a new trial, a local Dirichlet parameterized by $\alpha^L \in [1, 10]$ is initialized to $k\alpha^G$, where k is a normalizing factor. Such design of using a scaled local distribution is necessary to allow α^L to adapt faster than α^G within one trial; *i.e.*, agents must adapt more rapidly to the current trial compared to across all trials. Thus, we have a set of Dirichlet distributions to maintain beliefs: a Dirichlet for each attribute (*e.g.*, position, and color) as well as a Dirichlet for actions. Similarly, we maintain a Dirichlet distribution over each action a_i to encode beliefs regarding which actions are more likely to cause a state change, independent from any particular circumstance.

We introduce ρ to represent a causal event or observation occurring in the environment. Our agent wishes to assess the likelihood of a particular causal chain producing a causal event. The agent computes this likelihood by decomposing the chain into subchains

$$p(\rho|c; \beta) = \prod_{c_i \in c} p(\rho_i|c_i; \beta), \tag{1}$$

where $p(\rho_i|c_i; \beta)$ is formulated as

$$p(\rho_i|c_i; \beta) \propto p(\rho_i|a_i; \beta) \prod_{\substack{\phi_{ij} \in s_i \\ s_i \in c_i}} p(\rho_i|\phi_{ij}; \beta), \tag{2}$$

where $p(\rho_i|\phi_{ij}; \beta)$ and $p(\rho_i|a_i; \beta)$ follow multinomial distributions parameterized by a sample from the attribute and action Dirichlet distribution, respectively.[2] Intuitively, this bottom-up associative likelihood encodes a naive Bayesian prediction of how likely a particular subchain is to be involved with any causal event by considering how frequently the attributes and actions have been in causal events in the past, without regard for task structure. For example, we would expect an agent in OpenLock to learn that grey levers move under certain circumstances and white levers never move. This instance-level learning provides the agent with task-invariant, basic knowledge about which subchains are more likely to produce a causal effect.

3.2 Abstract-level Structure Learning

In this section, we outline how the agent learns abstract schemas; these schemas are used to encode generalized knowledge about task structure that is invariant to a specific observational environment.

A space of atomic causal schemas, Ω_{g^M}, of causal chain, CC, and CE, serve as categories for the Bayesian prior. The belief in each atomic schema is modeled as a multinomial distribution, whose parameters are defined by a Dirichlet distribution. This root Dirichlet distribution's parameters are updated after every trial according to the top-down causal theory γ, computed as the minimal graph edit distance between an atomic schema and the trial's solution structure.

---

[2]See supplementary materials for additional details.

This process yields a prior over atomic schemas, denoted as $p(g^M; \gamma)$, and provides the prior for the top-down inference process. Such abstraction allows agents to transfer beliefs between the abstract notions of CC and CE without considering task-specific requirements; *e.g.*, 3- or 4-lever configurations.

Next, we compute the belief in abstract instantiations of the atomic schemas. These abstract schemas share structural properties with atomic schemas but have a structure that matches the task definition. For instance, each schema must have three subchains to account for the 3-action limit imposed by the environment and should have N trajectories, where N is the number of solutions in the trial. Each abstract schema is denoted as g^A, and the space of abstract schemas, denoted Ω_{g^A}, is enumerated. The belief in an abstract causal schema is computed as

$$p(g^A; \gamma) = \sum_{g^M \in \Omega_{g^M}} p(g^A|g^M)p(g^M; \gamma). \tag{3}$$

The abstract structural space can be used to transfer beliefs between rooms; however, we need to perform inference over settings of positions and colors *in this trial* as the agent executes. Thus, the agent enumerates a space of instantiated schemas Ω_{g^I}, where each g^I is an instantiated schema. The agent then computes the belief in an instantiated schema as

$$p(g^I|do(q); \gamma) = \sum_{g^A \in \Omega_{g^A}} p(g^I|g^A, do(q))p(g^A; \gamma), \tag{4}$$

where $do(q)$ represents the *do* operator (Pearl 2009), and q represents the solutions already executed. Conditioning on $do(q)$ constrains the space to have instantiated solutions that contain the solutions already discovered by the agent in this trial. Causal chains c define the next lower level in the hierarchy, where each chain corresponds to a single attempt. The belief in a causal chain is computed as

$$p(c|do(q); \gamma) = \sum_{g^I \in \Omega_{g^I}} p(c|g^I, do(q))p(g^I|do(q); \gamma). \tag{5}$$

Finally, the agent computes the belief in each possible subchain as

$$p(c_i|do(\tau, q); \gamma) = \sum_{c \in \Omega_C} p(c_i|c, do(\tau, q))p(c|do(q); \gamma), \tag{6}$$

where $do(\tau, q)$ represents the intervention of performing the action sequence executed thus far in this attempt τ, and performing all solutions found thus far q. This hierarchical process allows the agent to learn and reason about abstract task structure, taking into consideration the specific instantiation of the trial, as well as the agent's history within this trial.[2]

Additionally, if the agent encounters an action sequence that does not produce a causal event, the agent prunes all chains that contain the action sequence from Ω_C and prunes all instantiated schemas that contain the corresponding chain from Ω_{g^I}. This pruning strategy means the agent assumes the environment is deterministic and updates its theory about which causal chains are causally plausible through interactions on-the-fly.

4 Intervention Selection

Our agent's goal is to pick the action it believes has the highest chance of (i) being causally plausible in the environment *and* (ii) being part of the solution to the task. We decompose each subchain c_i into its respective parts, $c_i = (a_i, s_i, cr_i^a, cr_i^s)$. The agent combines the top-down and bottom-up processes into a final subchain posterior:

$$p(c_i|\rho_i, do(\tau, q); \gamma, \beta) \propto p(\rho_i|c_i; \beta)p(c_i|do(\tau, q); \gamma). \quad (7)$$

Next, the agent marginalizes over causal relations and states to obtain a final, action-level term to select interventions:

$$p(a_i|\rho_i, do(\tau, q); \gamma, \beta) = \sum_{s_i \in \Omega_S} \sum_{cr_i^a \in \Omega_{CR}} \sum_{cr_i^s \in \Omega_{CR}} p(a_i, s_i, cr_i^a, cr_i^s|\rho_i, do(\tau, q); \gamma, \beta). \quad (8)$$

The agent uses a model-based planner to produce action sequences capable of opening the door (following human participant instructions in (Edmonds et al. 2018)). The goal is defined as reaching a particular state s^*, and the agent seeks to execute the action a_t to maximize the posterior subject to the constraints that the action appears in the set of chains that satisfy the goal, $\Omega_{C^*} = \{c \in \Omega_C \mid s^* \in c\}$. We define the set of actions that appear in chains satisfying the goal as $\Omega_{A^*} = \{a \in \Omega_A | \exists c \in \Omega_{C^*}, \exists\, s, cr^a, cr^s\, |(a, s, cr^a, cr^s) \in c\}$. The agent's final planning goal is

$$a_t^* = \arg\max_{a_i \in \Omega_{A^*}} p(a_i|\rho_i, do(\tau, q); \gamma, \beta). \quad (9)$$

At each time-step, the agent selects the action that maximizes this planning objective and updates its beliefs about the world as described in Section 3.1 and Section 3.2. This iterative process consists of optimal decision-making based on the agent's current understanding of the world, followed by updating the agent's beliefs based on the observed outcome.

5 Experiments

We compare results between predominate model-free RL algorithms with the proposed theory-based causal transfer model. Specifically, we compare the proposed method against Deep Q-Network (DQN) (Mnih et al. 2015), DQN with prioritized experience replay (DQN (PE)) (Schaul et al. 2016), Advantage Actor-Critic (A2C) (Mnih et al. 2016), Trust Region Policy Optimization (TRPO) (Schulman et al. 2015), Proximal Policy Optimization (PPO) (Schulman et al. 2017), and Model-Agnostic Meta-Learning (MAML) (Finn, Abbeel, and Levine 2017) agents. We use the term *positive transfer* and *negative transfer* to indicate that agent performance benefits from or is hindered by the training phase, respectively.

5.1 Experimental Setup

The proposed model follows the same procedure as the one used for human studies presented in Edmonds et al. 2018. Baseline (no transfer) agents are placed in 4-lever scenarios for all trials. Transfer agents are evaluated in two phases: training and transfer. For every training trial, the agent is placed into a 3-lever trial and allowed 30 attempts to find *all* solutions. In the transfer phase, the agent is tasked with a 4-lever trial. Critically, the agent only sees each trial (room) one

time, so generalizations must be formed quickly to transfer between trials successfully. See Section 2 for more details.

When executing various model-free RL agents under this experimental setup, no meaningful learning takes place. Instead, we train RL agents by looping through all rooms repeatedly (thereby seeing each room multiple times). Agents are also allowed 700 attempts in each trial to find all solutions. During training, agents execute for 200 training iterations, where each iteration consists of looping through all six 3-lever trials. During transfer, agents execute for 200 transfer iterations, where each iteration consists of looping through all five 4-lever trials. Note that the setup for RL agents is advantageous; in comparison, both the proposed model and human subjects are only allowed 30 attempts (versus 700) during the training and 1 iteration (versus 200) for transfer.

RL agents operate directly on the state of the simulator encoded as a 16-dimensional binary vector: (i) the status of each of the 7 levers (*pushed* or *pulled*), (ii) the color of each of the 7 levers (*grey* or *white*), (iii) the status of the door (*open* or *closed*) and (iv) the status of the door lock indicator (*locked* or *unlocked*). The 7-dimensional encoding of the status and color of each lever encodes the position of each lever; *e.g.*, the 0-th index corresponds to the upper-right position. Despite direct access to the simulator's state, RL approaches were unable to form a transferable task abstraction.

Additionally, we utilized a plethora of reward functions to explore under what circumstances these RL approaches may succeed. Our agents used sparse reward functions, shaped reward functions, and conditional reward functions that encourage agents to find unique solutions.[3] A reward function that only rewards for unique solutions performed best, meaning agents were only rewarded the *first* time they found a particular solution. This is similar to the human experimental setup, under which participants were informed when they found a solution for the first time (thereby making progress towards the goal of finding *all* solutions) but were not informed they executed the same solution multiple times (thereby not making progress towards the goal).

5.2 Reinforcement Learning Results

The model-free RL results, shown in Fig. 4, demonstrate that A2C, TRPO, and PPO are capable of learning how to solve the OpenLock task from scratch. However, A2C in the CC4 condition is the only agent showing positive transfer; every other agent in every condition shows negative transfer.

These results indicate that current model-free RL algorithms are capable of learning how to achieve this task; however, the capability to transfer the learned abstract knowledge is markedly different compared to human performance in Edmonds et al. 2018. Due to the overall negative transfer trends shown by nearly every RL agent, we conclude that these RL algorithms cannot capture the correct abstractions to transfer knowledge between the 3-lever training phase and the 4-lever transfer phase. Note that the RL algorithms found the CE4 condition more difficult than CC4, a result also shown in our proposed model results and human participants.

---

[3]See supplementary materials for the numerous architectures, parameters, and reward functions used.
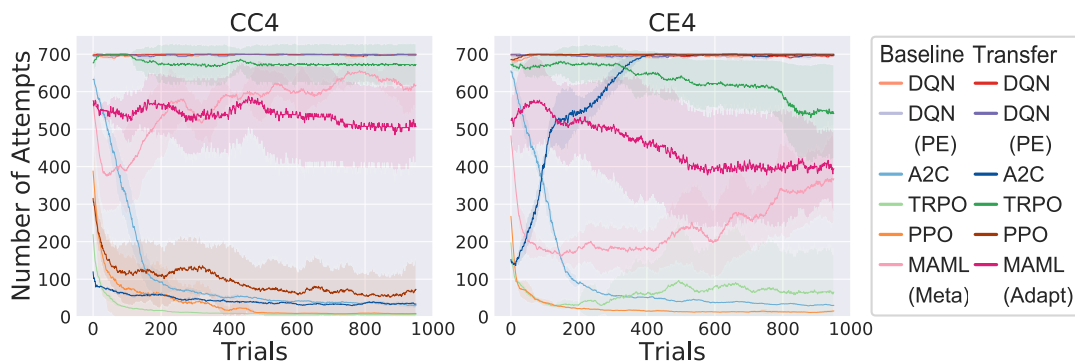
Figure 4: RL results for baseline and transfer conditions. Baseline (no transfer) results show the best-performing algorithms (PPO, TRPO) achieving approximately 10 and 25 attempts by the end of the baseline training for CC4 and CE4, respectively. A2C is the only algorithm to show positive transfer; A2C performed better with training for the CC4 condition. The last 50 iterations are not shown due to the use of a smoothing function.

5.3 Theory-based Causal Transfer Results

The results using the proposed model are shown in Fig. 5. These results are qualitatively and quantitatively similar to the human participant results presented in Edmonds et al. 2018, and starkly different from the RL results. We execute 40 agents in each condition, matching the number of human subjects described in Edmonds et al. 2018.

Our agent does not require looping over trials multiple times; it is capable of learning and generalizing from seeing each trial only one time. In the baseline agents, the CE4 condition was more difficult than CC4; this trend was also observed in human participants. During transfer, we see a similar performance as the baseline results; however, for congruent cases (transferring from the same structure with an additional lever) were easier than incongruent cases (transferring to a different structure with an additional lever; CE4 transfer); this result was statistically significant for CE4: $t(79) = 3.0$; $p = 0.004$. For CC4 transfer, no significance was observed ($t(79) = 0.63$; $p = 0.44$), indicating both CC3 and CE3 obtained near-equal performance when transferred to CC4.

These learning results are significantly different from the RL results; the proposed causal theory-based model is capable of learning the correct abstraction using instance and structural learning schemes, showing similar trends as the human participants. It is worth noting that RL agents were trained under highly advantageous settings. RL agents: (i) were given more attempts per trial; and (ii) more importantly, were allowed to learn in the same trial multiple times. In contrast, the present model learns the proper mechanisms to: (i) transfer knowledge to structurally equivalent but observationally different scenarios (baseline experiments); (ii) transfer knowledge to cases with structural differences (transfer experiments); and (iii) do so using the *same experimental setup* as humans. The model achieves this by understanding which scene components are capable of inducing state changes in the environment while leveraging overall task structure.[4]

---

[4] For additional model results and ablations, see supplementary.

6 Conclusion and Discussion

In this work, we show how the theory-based causal transfer coupled with an associative learning scheme can be used to learn transferable structural knowledge under both observationally and structurally varying tasks. We executed a plethora of model-free RL algorithms, none of which learned a transferable representation of the OpenLock task, even under favorable baseline and transfer conditions. In contrast, the proposed model results are not only capable of successfully completing the task, but also adhere closely to the human participant results in Edmonds et al. 2018.

These results suggest that current model-free RL methods lack the necessary learning mechanisms to learn generalized representations in hierarchical, structured tasks. Our model results indicate human causal transfer follows similar abstractions as those presented in this work, namely learning abstract causal structures and learning instance-specific knowledge that connects this particular environment to abstract structures. The model presented here can be used in any reinforcement learning environment where: (i) the environment is governed by a causal structure, (ii) causal cues can be uncovered from interacting with objects with observable attributes, and (iii) different circumstances share some common causal properties (structure and/or attributes).

6.1 Discussion

Why is causal learning important for RL? We argue that causal knowledge provides a succinct, well-studied, and well-developed framework for representing cause and effect relationships. This knowledge is invariant to extrinsic rewards and can be used to accomplish many tasks. In this work, we show that leveraging abstract causal knowledge can be used to transfer knowledge across environments with similar structure but different observational properties.

How can RL benefit from structured causal knowledge? Model-free RL is apt at learning a representation to maximize a reward within simple, non-hierarchical environments using a greedy process. Thus, current approaches do not restrict or impose learning an abstract structural representation of the environment. RL algorithms should be augmented with
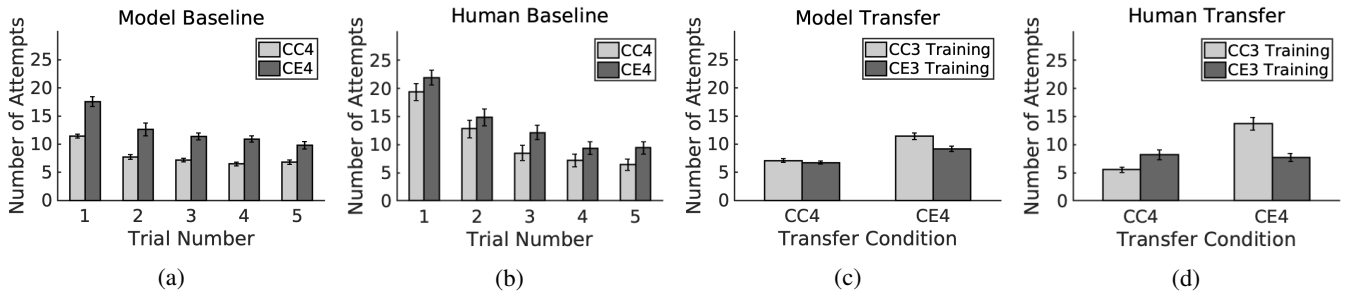
Figure 5: Model performance *vs.* human performance. (a) Proposed model baseline results for CC4/CE4. We see an asymmetry between the difficulty of CC and CE. (b) Human baseline performance (Edmonds et al. 2018). (c) Proposed model transfer results for training in CC3/CE3. The transfer results show that transferring to an incongruent CE4 condition (*i.e.*, different structure, additional lever; *i.e.*, CC3 to CE4) was more difficult than transferring to a congruent condition (*i.e.*, same structure, additional lever; *i.e.*, CE3 to CE4). However, the agent did not show a significant difference in difficulty when transferring to congruent or incongruent condition for the CC4 transfer condition. (d) Human transfer performance (Edmonds et al. 2018).

mechanisms to learn explicit structural knowledge and jointly optimized to learn both an abstract structural encoding of the task while maximizing rewards.

Why is CE more difficult than CC? Human participants, RL, and the proposed model all found CE more difficult than CC. A natural question is: why? We posit that it occurs from a decision-tree perspective. In the CC condition, if the agent makes a mistake on the first action, the environment will not change, and the rest of the attempt is bound to fail. However, should the agent choose the correct grey lever, the agent can choose either remaining grey levers; both of which will unlock the door. Conversely, in the CE condition, the agent has two grey levers to choose from in the first action; both will unlock the lever needed to unlock the door. However, the second action is more ambiguous. The agent could choose the correct lever, but it could also choose the other grey lever. Such complexity leads to more failure paths from a decision-tree planning perspective. The CC condition receives immediate feedback on the first action as to whether or not this plan will fail; the CE condition, on the other hand, has more failure pathways. We plan to investigate this property further, as this asymmetry was unexpected and unexplored in the literature.

What other theories may be useful for learning causal relationships? In this work, we adhere to an associative learning theory. We adopt the theory that *causal relationships induce state changes*. However, other theories may also be appealing. For instance, the associative theory used does not directly account for long-term relationships (delayed effects). More complex theories could potentially account for delayed effects; *e.g.*, when an agent could not find a causal attribute for a particular event, the agent could examine attributes jointly to best explain the causal effect observed. Prior work has examined structural analogies (Hinrichs and Forbus 2011; Zhang et al. 2019a; 2019b) and object mappings (Fitzgerald, Goel, and Thomaz 2018) to facilitate transfer; these may also be useful to acquire transferable causal knowledge.

How can hypothesis space enumeration be avoided? Hypothesis space enumeration can quickly become intractable as problems increase in size. While this worked used a fixed, fully enumerated hypothesis space, future work will include

examining how sampling-based approaches can be used to iteratively generate causal hypotheses. Bramley et al. 2017 showed a Gibbs-sampling based approach; however, this sampling should be guided with top-down reasoning to guide the causal learning process by leveraging already known causal knowledge with proposed hypotheses.

How well would model-based RL perform in this task? Model-based RL may exhibit faster learning within a particular environment but still lacks mechanisms to form abstractions that enable human-like transfer. This is an open research question, and we plan on investigating how abstraction can be integrated with model-based RL methods.

How is this method different from hierarchical RL? Typically, hierarchical RL is defined on a hierarchy of goals, where subgoals represent *options* that can be executed by a high-level planner (Chentanez, Barto, and Singh 2005). Each causally-plausible hypothesis can be seen as an option to execute. This work seeks to highlight the importance of leveraging causal knowledge to form a world-model and using said model to guide a reinforcement learner. In fact, our work can be recast as a form of hierarchical model-based RL.

Future work should primarily focus on how to integrate the proposed causal learning algorithm directly with reinforcement learning. An agent capable of integrating causal learning with reinforcement learning could generalize world dynamics (causal knowledge) and goals (rewards) to novel but similar environments. One challenge, unaddressed in this paper, is to how to generalize rewards to varied environments. Traditional reinforcement learning methods, such as Q-learning, do not provide a mechanism to extrapolate internal values to similar but different states. In this work, we showed how extrapolating causal knowledge can aid in uncovering the causal relationships in similar environments. Adopting a similar scheme for some form of reinforcement learning would enable reinforcement learners to succeed in the OpenLock task without iterating over the trials multiple times, and could enable one-shot reinforcement learning. Future work will also examine how a learner can iteratively grow a causal hypothesis while incorporating a background theory of causal relationships.

Acknowledgement

References

Bramley, N. R.; Dayan, P.; Griffiths, T. L.; and Lagnado, D. A. 2017. Formalizing Neurath's ship: Approximate algorithms for online causal learning. *Psychological Review* 124(3):301.

Bramley, N. R.; Gerstenberg, T.; Tenenbaum, J. B.; and Gureckis, T. M. 2018. Intuitive experimentation in the physical world. *Cognitive Psychology* 105:9–38.

Bramley, N. R.; Lagnado, D. A.; and Speekenbrink, M. 2015. Conservative forgetful scholars: How people learn causal structure through sequences of interventions. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 41(3):708.

Cheng, P. W. 1997. From covariation to causation: a causal power theory. *Psychological Review* 104(2):367.

Chentanez, N.; Barto, A. G.; and Singh, S. P. 2005. Intrinsically motivated reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Edmonds, M.; Kubricht, J.; Summers, C.; Zhu, Y.; Rothrock, B.; Zhu, S.-C.; and Lu, H. 2018. Human causal transfer: Challenges for deep reinforcement learning. In *Annual Meeting of the Cognitive Science Society (CogSci)*.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 1126–1135. JMLR. org.

Fitzgerald, T.; Goel, A.; and Thomaz, A. 2018. Human-guided object mapping for task transfer. *ACM Transactions on Human-Robot Interaction (THRI)* 7(2):17.

Griffiths, T. L., and Tenenbaum, J. B. 2005. Structure and strength in causal induction. *Cognitive Psychology* 51(4):334–384.

Griffiths, T. L., and Tenenbaum, J. B. 2009. Theory-based causal induction. *Psychological Review* 116(4):661–716.

Heider, F. 1958. *The Psychology of Interpersonal Relations*. Psychology Press.

Hinrichs, T., and Forbus, K. D. 2011. Transfer learning through analogy in games. *AI Magazine* 32(1):70–70.

Holyoak, K., and Cheng, P. W. 2011. Causal learning and inference as a rational process: The new synthesis. *Annual Review of Psychology* 62:135–163.

Kansky, K.; Silver, T.; Mély, D. A.; Eldawy, M.; Lázaro-Gredilla, M.; Lou, X.; Dorfman, N.; Sidor, S.; Phoenix, S.; and George, D. 2017. Schema networks: Zero-shot transfer with a generative causal model of intuitive physics. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 1809–1818. JMLR. org.

Legg, S., and Hutter, M. 2007. Universal intelligence: A definition of machine intelligence. *Minds and Machines* 17(4):391–444.

Levine, S.; Finn, C.; Darrell, T.; and Abbeel, P. 2016. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research* 17(1):1334–1373.

Lillicrap, T. P.; Hunt, J. J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; and Wierstra, D. 2015. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*.

Maclaurin, C. 1742. *A Treatise of Fluxions: In Two Books. 1*, volume 1. Ruddimans.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A. K.; Ostrovski, G.; et al. 2015. Human-level control through deep reinforcement learning. *Nature* 518(7540):529.

Mnih, V.; Badia, A. P.; Mirza, M.; Graves, A.; Lillicrap, T.; Harley, T.; Silver, D.; and Kavukcuoglu, K. 2016. Asynchronous methods for deep reinforcement learning. In *International Conference on Machine Learning (ICML)*.

Newton, I., and Colson, J. 1736. *The Method of Fluxions and Infinite Series; with Its Application to the Geometry of Curve-lines*. Henry Woodfall; and sold by John Nourse.

Pearl, J. 2009. *Causality*. Cambridge University Press.

Rescorla, R. A., and Wagner, A. R. 1972. A theory of pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. *Classical conditioning II: Current research and theory* 2:64–99.

Schaul, T.; Quan, J.; Antonoglou, I.; and Silver, D. 2016. Prioritized experience replay. In *International Conference on Learning Representations (ICLR)*.

Schulman, J.; Levine, S.; Abbeel, P.; Jordan, M.; and Moritz, P. 2015. Trust region policy optimization. In *International Conference on Machine Learning (ICML)*.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Shanks, D. R., and Dickinson, A. 1988. Associative accounts of causality judgment. *Psychology of learning and motivation* 21:229–261.

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. 2016. Mastering the game of go with deep neural networks and tree search. *Nature* 529(7587):484–489.

Stahl, A. E., and Feigenson, L. 2015. Observing the unexpected enhances infants' learning and exploration. *Science* 348(6230):91–94.

Tenenbaum, J. B.; Griffiths, T. L.; and Kemp, C. 2006. Theory-based bayesian models of inductive learning and reasoning. *Trends in Cognitive Sciences* 10(7):309–318.

Thielscher, M. 1998. *Introduction to the fluent calculus*. Citeseer.

Waldmann, M. R., and Holyoak, K. J. 1992. Predictive and diagnostic learning within causal models: Asymmetries in cue competition. *Journal of Experimental Psychology: General* 121(2):222–236.

Zhang, C.; Vinyals, O.; Munos, R.; and Bengio, S. 2018. A study on overfitting in deep reinforcement learning. *arXiv preprint arXiv:1804.06893*.

Zhang, C.; Gao, F.; Jia, B.; Zhu, Y.; and Zhu, S.-C. 2019a. Raven: A dataset for relational and analogical visual reasoning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhang, C.; Jia, B.; Gao, F.; Zhu, Y.; Lu, H.; and Zhu, S.-C. 2019b. Learning perceptual inference by contrasting. In *Advances in Neural Information Processing Systems (NeurIPS)*.