The Difficulty of Novelty Detection and Adaptation in Physical Environments

Vimukthini Pinto[1][0000−0003−2693−8198], Chathura Gamage[1], Matthew Stephenson[2], and Jochen Renz[1]

[1] The Australian National University, Canberra, Australia
[2] Flinders University, Adelaide, Australia
vimukthini.inguruwattage@anu.edu.au

Abstract. Detecting and adapting to novel situations is a major challenge for AI systems that operate in open-world environments. One reason for this challenge is due to the diverse range of forms that novelties can take. To accurately evaluate an AI system's ability to detect and adapt to novelties, it is crucial to investigate and formalize the difficulty of different novelty types. In this paper, we propose a method for quantifying the difficulty of novelty detection and novelty adaptation in open-world physical environments, considering factors such as the appearance and location of objects, as well as the actions required by the agent. We implement several difficulty measures using a combination of qualitative spatial relations, learning algorithms, and statistical distance measures. To demonstrate an application of our approach, we apply our difficulty measures to novelties in the popular physics simulation game Angry Birds. We invite researchers to incorporate the proposed novelty difficulty measures when evaluating AI systems to gain a better understanding of their limitations and identify areas for future improvement.

Keywords: AI Evaluation · Difficulty · Novelty · Open-world Learning

1 Introduction

Autonomous AI systems such as self-driving cars, space probes, and surveillance drones have become increasingly popular and common in recent years. These AI systems require the ability to detect and adapt to novel situations in a timely and efficient manner to avoid undesirable consequences. For instance, if a self-driving car maintains its speed in a storm that was not experienced during model training, it could endanger many lives. Open-world learning (OWL) is an emerging field of study that aims to solve the challenge of detecting and adapting to novel situations [14]. To progress in OWL research, it is essential to have appropriate evaluation protocols to capture the performance of agents under the two tasks: novelty detection and adaptation [18,9]. This paper contributes to the OWL evaluation by creating difficulty measures to independently evaluate agents' performance from the inherent difficulty of novelties.

We encounter a near-infinite form of novelties in the real world [14,1,6]. For example, consider an autonomous car designed for urban driving in a busy city. The model which controls the vehicle has reliable expertise for navigating in this setting, but suppose it enters a new area with different traffic patterns. Here, it may encounter new types of road signs, unfamiliar pedestrians that can cross its path, and aggressive drivers that may cut it off. The vehicle may enter a stormy area where visibility is low and where sensor readings are distorted or where strong winds threaten to push it off course. As seen from this example, there is a large range of novelties and some of them may be easy to detect and adapt and some of them could be hard or nearly impossible. If we are to evaluate the novelty detection and novelty adaptation ability of an agent, it would be uninformative to comment on the performance by considering all the novelties as a whole [19]. Using a measure of difficulty in the evaluation enables evaluators to understand the range of situations an agent may fail and reliably make conclusions.

In this paper, we identify different forms of novelties by considering two states where an agent can detect/adapt to novelty: *observational state*, *action state*. The observational state considers the situations that can be visually perceived while the action state considers the actions an agent has to take. We formalize practical methods to compute difficulty under the two states with the use of learning algorithms and statistical distance measures. We utilize existing learning algorithms and we propose an algorithm developed using qualitative spatial relations (QSRs). For statistical distance measures, we use graph edit distance (GED), distance measures developed using solution paths, and measures based on the probability density function (PDF). In the supplementary materials, we demonstrate that the difficulty measures we formulated can be easily applied in practice by applying them to a recently developed testbed NovPhy [9] (Angry Birds with novelty), which injects novelties into a physical environment.

2 Background and Related Work

In this section, we first provide definitions for the terms that we will be using throughout the paper. Next, we review the literature on a number of topics required to understand our novelty difficulty measures.

- *novelty*: a situation that an agent has not encountered during model training. It could be a new object that an agent has not seen before or a phenomenon that an agent has not experienced (eg: storms, floods).
- *pre-novelty*: a situation without novelty (i.e., a situation that an agent has seen during model training).
- *novel object*: An object that has one or more novel properties. It could be an object that an agent has seen before but with a different color, or mass, or the object may do an action that it did not do during pre-novelty.
- *non-novel object*: An object without any novel property.
- *object class*: A group of objects with similar properties. (eg: In Fig. 1b, there are multiple brown circular objects. They belong to class *wood-circle*).
- *novelty detection*: detecting that a novelty is present in a task.
- *novelty adaptation*: solving a task in the presence of a novelty.

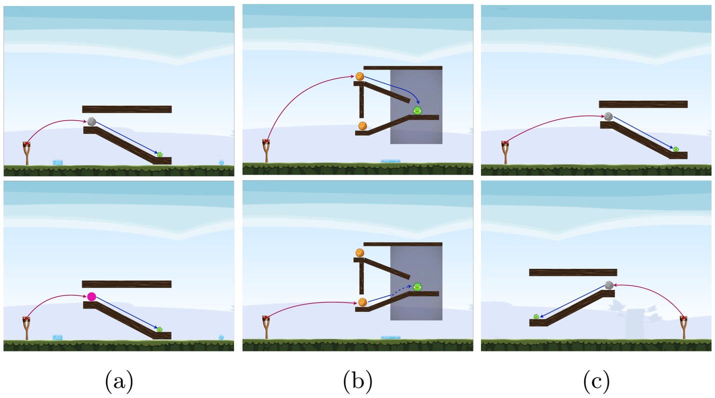


Fig. 1: Example tasks from NovPhy testbed. In each subfigure, the top figure is the pre-novelty task and the bottom figure is the corresponding task with the novelty. The arrows show the trajectories of the objects when the solution is executed. It can be seen that in (a), we do not need to modify the shooting angle (action) as the novelty only changes the colour of an object, in contrast, in (b) and (c) we need to change the action due to the nature of the novelty. See [9] for descriptions of the novelties.

2.1 Novelty Research

In recognition of the critical need for AI systems that can effectively detect and adapt to novel situations, DARPA has launched a program known as the Science of Artificial Intelligence and Learning for Open-world Novelty (SAIL-ON) The SAIL-ON program defines novelty in the realm of AI as situations that violate implicit or explicit assumptions about the agents, the environment, or their interactions [20]. [1] also look into a range of novelties and formalize a theory of novelty for open-world environments. The authors identified three distinct states where novelties can occur, *observational state*, *world state*, and *agent state*. Our work on defining the difficulty of detection and adaptation follows from this research and we consider the observational state where a novelty can be visually perceived. However, we do not consider the world state (the state where all the information is available. eg: physical parameter values in a physical domain) as our work focuses on detection and adaptation difficulty and an agent does not receive all the information from the world state. We also do not consider the agent state (a state specific to an agent based on the agent architecture) as we intend to develop difficulty measures that do not depend on individual agent properties. Instead, we consider *action state* that takes into account the actions an agent needs to take to solve a task. In Fig. 1, we show example novelties where an agent needs to modify the action to solve the task and where agents do not need to modify the action.

2.2 Difficulty Prediction

Difficulty assessment is a popular research area in a number of research fields ranging from neuroscience to AI. In neuroscience, researchers study the difficulty of decision-making processes in humans [7,10]. Similarly, in AI, researchers study the difficulty of tasks for AI systems [11,16]. Our work on developing difficulty measures for novelty detection and adaptation focuses on AI agents.

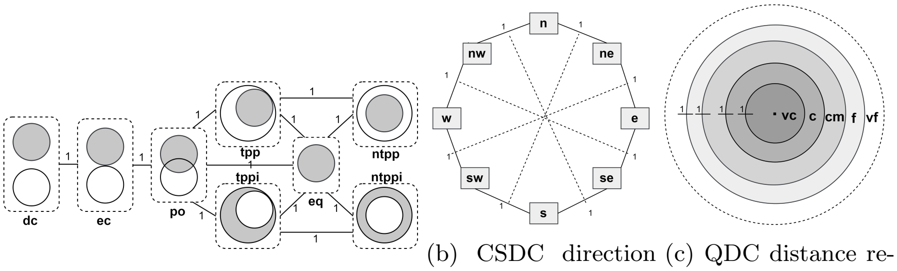
Considering the OWL-related difficulty measures already available in the literature, [19] propose a difficulty of detection only by considering a single class of novelties. The authors only consider novelties that cannot be visually perceived but are different in underlying physical parameters. Considering the adaptation difficulty, [17] propose a method to quantify the difficulty of adapting to novelty using the solution paths the agents take. The method requires a reinforcement learning agent to solve multiple tasks, which is costly as it requires multiple trial and error runs to reach the optimal solution. Inspired by this approach, we also propose a method that takes solution paths into consideration without running an agent. Another novelty adaptation difficulty measure predicts the difficulty using GED [21] for an agent's mental model for the board game Monopoly [13]. Similarly, we make use of GED and we propose an agent-independent difficulty measure for physical environments. [6], analyze domain complexity and introduce factors such as single entity and multiple entities that contribute towards difficulty in OWL tasks. Our work makes use of these factors when defining difficulty measures.

2.3 Learning Algorithms

Novelty detection, which is sometimes referred to as anomaly detection, outlier detection, or one-class classification, [2] is a critical research area in machine learning and data mining. The goal of novelty detection is to identify patterns or behaviors in data that are significantly different from the expected or normal behavior. Over the years, several comprehensive surveys and reviews of novelty detection techniques have been conducted, focusing on different aspects and techniques [12]. In [2], authors reviewed novelty detection techniques and categorized them into six domains: classification, clustering, nearest neighbors, statistical methods, information theory, and spectral theory. Our work on predicting the difficulty of novelty detection, uses classification-based approaches and clustering-based approaches. We identify different factors that contribute towards the detection such as color and shape and we have developed multiple classification models to predict the probability of an object belonging to different object classes. Finally, we combine the models in the form of a weighted ensemble to predict the difficulty. The clustering-based model is an algorithm we propose to predict the detection difficulty using QSRs.

2.4 Qualitative Spatial Relations (QSRs)

Due to a large number of applications of QSRs [3,5,15], dozens of formalisms of calculi have been proposed for describing various aspects of space [5,4,3]. In our work, we utilize three commonly used calculi to represent topology, direction, and distance. We use *Region Connection Calculus (RCC)-8* (Fig. 2a) to describe topological relations: *dc* (disconnected), *ec* (externally connected), *po*



(a) RCC-8 topology relations

(b) CSDC direction relations

(c) QDC distance relations

Fig. 2: QSRs. The connections represent conceptual neighbors. Distance between two neighbors is taken as 1.

(partially overlapping), *eq* (equal), *tpp* (tangential proper part), *tppi* (tangential proper part inverse), *ntpp* (non-tangential proper part), and *ntppi* (non-tangential proper part inverse) [4,3]. To describe directions, we use *Cone-Shaped Direction Calculus (CSDC)* (Fig. 2b). The relations of the CSDC are based on the eight disjoint sectors of the space divided by the lines going through the reference point [3]. The eight relations are *n* (north), *ne* (northeast), *e* (east), *se* (southeast), *s* (south), *sw* (southwest), *w* (west), and *nw* (northwest). Additionally, we use *Qualitative Distance Calculus (QDC)* (Fig. 2c) to describe distance between two objects. We have used five absolute distance calculi: *vc* (very close), *cl* (close), *cm* (commensurate), *fr* (far), and *vf* (very far) [3].

2.5 Experimental Domain

Our experimental domain NovPhy [9] is a testbed designed to evaluate physical reasoning in the presence of novelties. The testbed is based on the popular physics simulation game Angry Birds. Fig. 1 shows example tasks with and without novelty. We explain the experimental domain in detail in the Supplementary A along with the experimental results.

3 Novelty Difficulty Formulation

In this section, we explain the dimensions of novelty we need to consider to formulate our difficulty measures and the practical implementation of it.

3.1 Dimensions of Novelty

Considering the dimensions where novelty can be perceived by an agent, we consider two states: *observational state* and *action state*. The two states are motivated by the research [1] on the unifying theory of novelty.

1. *Observational state*: The state where an agent can observe the environment
2. *Action state*: The state where the agent needs to take an action

We consider these two states, as in a physical domain, the novelties can either be detected due to the physical appearance (novelties that can be perceived visually: in the observational state) or could be detected/adapted due to

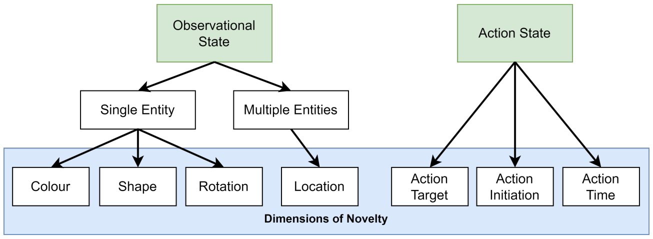


Fig. 3: The novelty dimensions considered in the study

a change in action compared to the actions encountered before (novelties that can be perceived after taking an action: in the action state). As [6] state, a novelty in an environment can be in a single entity or it could be as a relationship between multiple entities. Taking the entities into consideration, in the observational state, we identify dimensions that can be observed in a single entity and dimensions that can be observed as a relationship between multiple entities. Therefore, we extend the dimensions to color, shape, and rotation for single entity, and for multiple entities we consider the relative location of objects in terms of QSRs. Considering the action state, the dimensions we consider are based on situations that enable an agent to detect/adapt to novelty if an agent's expected action changes from the known action in a situation without novelty. We consider 1) *action target*, to check if the target object changes (for example, in Fig. 1b, the target object has changed when the direction of the air turbulence has changed), 2) *action initiation* to check if the solution action changes (for example, in Fig. 1c, the shooting angle has changed compared to the angle used in the non-novelty), and 3) *action time* to identify if the time allocated to take actions change (for example, in novelty you may need to shoot faster before a certain event happens). See Fig. 3 for the novelty dimensions we consider.

3.2 Observational State

In this section, we discuss the formulation of difficulty under each dimension belonging to the observational state. We discuss the formulation of color and shape together as the underlying formulation is the same except for the input data structure. For the rotation based difficulty measure, we make use of the predictions made from shape and color models. For the location, we use QSRs to develop our difficulty prediction algorithm. All the observational state difficulty measures are aimed to predict the difficulty of novelty detection. The measures are not used to predict the adaptation difficulty as agents cannot take any action in the observational state. Therefore, we predict the difficulty of novelty detection for agents if the agents are detecting novelty using a single dimension.

Color and Shape Based Difficulty Given a physical domain, there are non-novel objects that an agent can be trained on. Therefore, with the use of classification algorithms, we can predict the probability that a novel object belongs to a non-novel object class. The color-based algorithm predicts the probability that a novel object can be considered an object that was seen before based on the

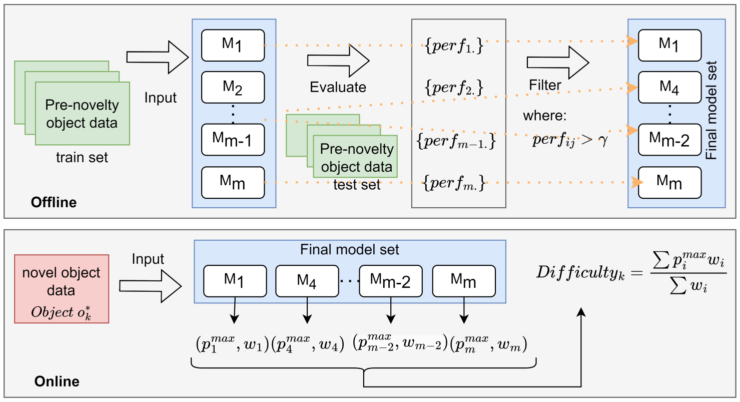


Fig. 4: The formulation of color and shape based difficulty

observed color of an object. Similarly, in the shape-based algorithm, we predict the probability that a novel object can be considered an existing object based on the shape of the object.

We illustrate the process of computing the difficulty based on color or shape in Fig. 4. There are two stages when computing difficulty for color and shape. In the offline stage, we train multiple learning models to classify objects and we evaluate the models on a pre-defined test set that comprises objects seen in pre-novelty. We define O as a list of all object classes in pre-novelty ($O = \{o_1, o_2, ..., o_j, ..., o_n\}$). Next, we filter out the models that have an acceptable performance. If the model M_i has a performance higher than an acceptable threshold γ for all object classes, we select the model to make predictions in the online stage ($Perf_{i,j} > \gamma \ \forall o_j \in O$). Therefore, the models that perform well will be used in the online stage in the form of a weighted ensemble to predict the probability (pr) that a novel object belongs to an existing object class. Following is the formulation to establish the difficulty value for novel object k, denoted o^*_k.

In model M_i: $P_{i,k} = \{p_{1,k}, p_{2,k}, ..., p_{j,k}, ..., p_{n,k}\}$, where, $p_{j,k} = pr(o^*_k = o_j)$. Therefore, we define $p^{max}_{i,k} = max(P_{i,k})$ and $\hat{o}_{i,k} = \arg\max_{o_j}(P_{i,k})$. i.e., $p^{max}_{i,k}$ is the maximum probability that the model M_i allocates o^*_k to an existing object class and $\hat{o}_{i,k}$ is the predicted object class. The weight of the prediction is calculated as $w_{i,k} = (\alpha n - n'_{i,k})/\alpha n$ if $\alpha n > n'_{i,k}$, or else $w_{i,k}$ is set to 0. Here, $n'_{i,k} = (\sum_{j=1}^{n} \begin{cases} 1 & if (p^{max}_{i,k} - p_{j,k}) \leq \beta \\ 0 & otherwise \end{cases}) - 1$. α and β are hyper-parameters that should be selected according to the domain based on the prediction flexibility we allow. The final difficulty of detecting o^*_k novel object using color/shape:

$$Difficulty^{color/shape}_k = \frac{\sum_{i=1}^{m} p^{max}_{i,k} w_{i,k}}{\sum_{i=1}^{m} w_{i,k}} \tag{1}$$

The difficulty value is between 0-1, 1 indicates the highest difficulty. The idea is that if a weighted ensemble predicts that a novel object belongs to an existing class based on its color/ shape, then it is difficult to visually detect. In contrast, the difficulty will be low if the models have lower probabilities or if there are multiple objects with probabilities close to maximum probability.

Rotation Based Difficulty Rotation difficulty measures if the agents can detect a novelty based on the rotation of the novel object (eg: in NovPhy, it could be that the pig is rotated upside down in novelty which is not usually rotated in pre-novelty). Similar to the previous section, the rotation difficulty also comprises two stages (See Fig.2 in Supplementary B). In the offline stage, we collect rotation data from non-novel object classes and estimate the distribution for each object class using kernel density estimation (KDE). In the online stage, we use the predicted object class from color and shape algorithms $\hat{o}_k$ and the novel objects' observed rotation (rot_k^*). The predicted object from color and shape algorithms can be selected based on a voting technique of choice (eg: hard voting, soft voting, or weighted voting) [22]. Next, the KDE of the predicted object is selected, and the area under the PDF for the $rot_k^* \pm rot_\epsilon$ can be interpreted as the difficulty of detection using rotation. rot_ϵ is a predefined constant (a small rotation shift) that helps to get the area under the probability density function ($PDF_r(\hat{o}_k)$). The difficulty based on rotation can be expressed as follows.

$$Difficulty_k^{rotation} = \int_{rot_k^*-rot_\epsilon}^{rot_k^*+rot_\epsilon} PDF_r(\hat{o}_k)\, dr \tag{2}$$

The underlying idea is that if the rotation of the novel object is commonly observed in pre-novelty, it becomes challenging to detect the novelty solely based on rotation. The rotation difficulty ranges between 0-1, 1 is the highest difficulty.

Location Based Difficulty The location-based difficulty considers the relative location between pairs of objects. For example, in Fig. 1c, the direction relationship between the bird and the pig has changed in novelty. The relative location between objects is captured through the change in QSRs between object pairs in novel tasks compared to the non-novel tasks. In the offline stage, we collect object pair relationships and develop clusters for each object pair based on a conceptual distance measure. In the online stage, we take the observed object pair relationship and determine the difficulty based on the distance to the clusters developed in the offline stage.

Formulation: To explain the difficulty measure, we first define a state as s_k^{ij} a tuple consisting of the classes of an object pair and its observed QSRs $(< o_i, o_j >, R_k^{ij})$ where $R_k^{ij} = [r_{1k}^{ij}, r_{2k}^{ij}, r_{3k}^{ij}]$. The set of all available states for the pair of object classes o_i and o_j is $S^{ij} = \{s_1^{ij}, s_2^{ij}, ...\}$. The relations $r_{1.}^{..} \in RCC\text{-}8$, $r_{2.}^{..} \in CSDC$, and $r_{3.}^{..} \in QDC$. Considering two states for the o_i and o_j object pair as s_k^{ij} and $s_{k'}^{ij}$, $s_k^{ij} = (< o_i, o_j >, R_k^{ij})$ and $s_{k'}^{ij} = (< o_i, o_j >, R_{k'}^{ij})$, we can find the distance between two states as $d_{s_k^{ij}, s_{k'}^{ij}} = \sum_{q=1}^{3} |r_{qk}^{ij} - r_{qk'}^{ij}|$. $|r_{qk}^{ij} - r_{qk'}^{ij}|$ is the minimum absolute distance between two relations that can be calculated using shortest path algorithms applied to the QSR graphs in Fig. 2.

Clustering: When constructing the clusters, we develop clusters for each object class pair. The cluster nodes are the QSRs (eg: R_k^{ij}) and two nodes R_k^{ij} and $R_{k'}^{ij}$ will be connected only if $d_{s_k^{ij}, s_{k'}^{ij}} \leq d^*$ (d^* is a threshold distance to define according to the domain to determine if the nodes connect). If $d_{s_k^{ij}, s_{k'}^{ij}} = 0$, the node size increases. The final size of the node R_k^{ij} is, $size(R_k^{ij}) = \frac{|\{k' | d_{s_k^{ij}, s_{k'}^{ij}} = 0\}|}{|S^{ij}|}$

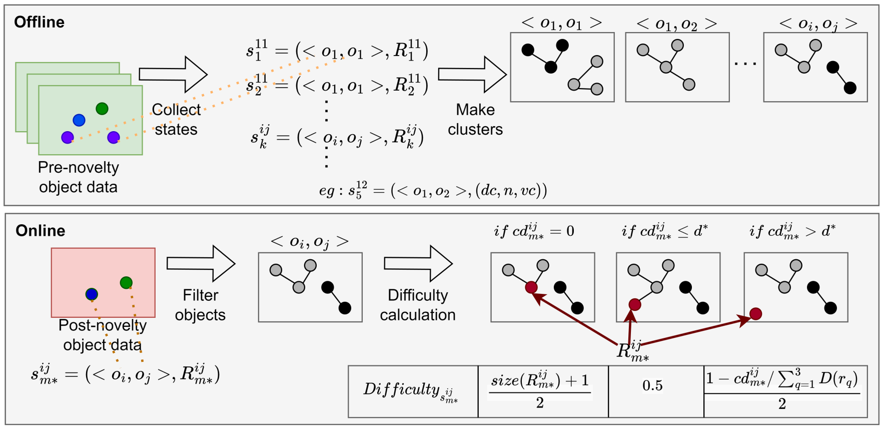

Offline
Pre-novelty object data
Collect states
$s_1^{11} = (< o_1, o_1 >, R_1^{11})$
$s_2^{11} = (< o_1, o_1 >, R_2^{11})$
$s_k^{ij} = (< o_i, o_j >, R_k^{ij})$
$eg : s_5^{12} = (< o_1, o_2 >, (dc, n, vc))$
Make clusters
$< o_1, o_1 >$
$< o_1, o_2 >$
$< o_i, o_j >$
Online
Post-novelty object data
$s_{m*}^{ij} = (< o_i, o_j >, R_{m*}^{ij})$
Filter objects
$< o_i, o_j >$
Difficulty calculation
$if\ cd_{m*}^{ij} = 0$
$if\ cd_{m*}^{ij} \leq d^*$
$if\ cd_{m*}^{ij} > d^*$
R_{m*}^{ij}
$Difficulty_{s_{m*}^{ij}}$
$\frac{size(R_{m*}^{ij})+1}{2}$
0.5
$\frac{1 - cd_{m*}^{ij}/\sum_{q=1}^{3} D(r_q)}{2}$

Fig. 5: The formulation of location based difficulty

(i.e., the proportion of states that has the same QSR as the state s_k^{ij}). We represent the set of clusters developed for S^{ij} as C^{ij} ($C^{ij} = \{c_1^{ij}, c_2^{ij}, ...\}$) where $c_l^{ij} = \{R_{l1}^{ij}, R_{l2}^{ij}, ...\}$.

Difficulty Computation: The difficulty computation will be done for each object pair in the novel task. Given an object pair (novel/ non-novel) with their observed QSRs and the object classes, the clusters developed in the offline stage for the corresponding object class pair will be extracted. Assume that the state we take from the novel task is s_{m*}^{ij}. Therefore, the set of clusters developed for S^{ij} will be extracted (i.e., C^{ij}). The cluster distance between C^{ij} and R_{m*}^{ij} is taken to be the minimum distance between the observed relations and relations available in the cluster set. That is $cd_{m*}^{ij} = min\{d_{s_{m*}^{ij}, s_k^{ij}} \forall s_k^{ij} \in S^{ij}\}$. We define the location difficulty for an observed state in equation 3 and illustrate the process in Fig. 5. The $D(r_1), D(r_2)$, and $D(r_3)$ in the equation represent the diameter of the *RCC-8*, *CSDC*, and *QDC* graphs (Fig. 2) respectively. The diameter of a graph is the length of the shortest path between the most distanced nodes.

$$Difficulty_{s_{m*}^{ij}}^{location} = \frac{1 + \begin{cases} size(R_{m*}^{ij}) & if cd_{m*}^{ij} = 0 \\ 0 & if cd_{m*}^{ij} \leq d^* \\ -(cd_{m*}^{ij})/(\sum_{q=1}^{3} D(r_q)) & otherwise \end{cases}}{2} \tag{3}$$

The rationale of this measure is that if the QSR of the pair of objects being examined belongs to a cluster node ($cd_{m*}^{ij} = 0$), it indicates that the QSR has been observed in pre-novelty. Thus, we quantify the difficulty based on the node size of the QSR (high difficulty). If $cd_{m*}^{ij} < d*$, QSR can be connected to an existing cluster and the difficulty is 0.5. If it cannot be connected, then we quantify using the distance away as a proportion of the maximum allowed distance. Similar to other measures, difficulty is between 0-1, 1 is the highest difficulty.

Observational state difficulty summary All the difficulty measures developed for the observational state are for detection and they are formulated for a single object (a single object pair for location). However, there are multiple objects in a task. i.e., there can be multiple novel objects in a task that should be considered for color, shape, and rotation-based measures. For the location-based measure, we should consider all the pairwise relations available. When defining the difficulty measure for a task (for each novelty dimension), we take the minimum of the difficulty values. For example, if a task contains multiple novel objects but only one object with a drastic difference in color compared to the non-novel objects, this object will have a lower difficulty of detection using color, therefore, we take the minimum difficulty as the task difficulty under the color dimension. Formally we define the task difficulty for novelty-dimension nd as $Difficulty^{nd} = min(Difficulty_k^{nd})$ for all novel objects/object pairs.

3.3 Action State

In this section, we present the formulation of difficulty measures for each dimension pertaining to the action state. An action within a given environment represents a point in the action space, such as selecting the shooting angle in Angry Birds. For each novelty dimension, we derive two difficulty measures: novelty detection and novelty adaptation. To develop these difficulty measures, we use the novel task template and the corresponding non-novel task template and we compare the action change between them.

Action Target Action target based difficulty applies to environments that have target objects to solve the tasks. This difficulty measure assesses if the target object changes between the novel task and the non-novel task. For example, in Fig. 1b, the non-novel task requires targeting the top ball, while the novel task requires targeting the bottom ball to solve the task due to the change in air turbulence. The adaptation difficulty in this task is high as an agent needs to change the target object. However, as the target object changes, the detection difficulty would be low as agents can detect that there is a novelty when the previous target object fails to solve the task. Therefore, we define action target difficulty using the *GED*. We define the action target graph for the non-novel task as $G_{non-novel}$ and for the novel task as G_{novel}. We represent the set of nodes of G_x as G_x^{nodes} where $x \in$ *{non-novel, novel}*. The edges indicate if any of the connected nodes is a target of the other node. Therefore, the total possible edits of G_x is $T_x^{edits} = |G_x^{nodes}| + \binom{|G_x^{nodes}|}{2}$. i.e., the total number of nodes and the total number of edges of a fully connected graph (See Supplementary Fig. 3a). Thus, the adaptation difficulty measure is defined as the ratio of necessary edits (to change G_{non_novel} to G_{novel}) to the total possible edits.

$$Difficulty_{adaptation}^{action\ target} = \frac{GED(G_{non-novel}, G_{novel})}{max(T_{non-novel}^{edits}, T_{novel}^{edits})} \tag{4}$$

Action Initiation The action initiation based difficulty evaluates if the action that leads to the solution differs between the novel task and the non-novel task. The underlying intuition is that if the novel task has solutions that an agent can learn in the non-novel task, it is easy to adapt (lower adaptation difficulty) but would be difficult to detect as the task gets solved by the same actions taken in the non-novel task (higher detection difficulty). The solution to solve a task can be defined according to the domain. For example, in Angry Birds, it would be defined as the shooting angle. We define S_x as the set of solutions for task x where $x \in$ *{non-novel, novel}* (Illustrated in Supplementary Fig.3b).

$$Difficulty_{adaptation}^{action\ initiation} = \frac{|S_{non-novel} \cap S_{novel}|}{|S_{novel}|} \tag{5}$$

Action Time The action time based difficulty looks at the time restrictions imposed on the novel task. For example, if the novel task requires an agent to take an action faster than that of the time allocated in the non-novel task, the adaptation difficulty would be higher. In contrast, as the task cannot be solved if an agent did not take the action, it would be easy for the agent to detect. Therefore we look at the proportion of time allocated to the novel task compared to the non-novel task when defining the action time based difficulty. In this formulation, we assume that a novel task cannot be allocated more time than a non-novel task. $time_x$ is the time allocated for the task x, where $x \in$ *{non-novel, novel}* (Illustrated in Supplementary Fig.3c).

$$Difficulty_{adaptation}^{action\ time} = \frac{time_{non-novel} - time_{novel}}{time_{non-novel}} \tag{6}$$

Action state difficulty summary The aforementioned difficulty measures, based on the novelty dimensions for the action state are formulated to assess novelty adaptation. As previously explained, the detection difficulty is the inverse of the adaptation difficulty. For each novelty dimension $nd \in$ *{action target, action initiation, action time}*, we define the difficulty of novelty detection as:

$$Difficulty_{detection}^{nd} = 1 - Difficulty_{adaptation}^{nd} \tag{7}$$

4 Discussion and Conclusion

The novelties that appear in OWL environments may take various forms and the difficulty to detect them and to adapt to them vary. While previous studies have not explored the impact of different novelty dimensions on difficulty, it is a crucial aspect to consider for conducting fair evaluations. Thus, our paper proposed pragmatic methods to evaluate the difficulty of novelties by considering a range of novelty dimensions and using a range of evaluation techniques inspired by statistical distance measures, learning techniques, and QSRs. In the supplementary we show how the difficulty measures can be applied in practice to analyse agents. Our difficulty formulations enable us to conduct a comprehensive evaluation by disentangling the difficulty of novelty with the performance. Moreover, our difficulty formulations can be embedded as a component to novelty generators [13,8] to generate tasks with a predefined difficulty.

We aim to expand this study by incorporating additional novelty dimensions such as dimensions to capture spatiotemporal changes. Moreover, we plan to conduct an evaluation of each novelty dimension by creating novelties that consider variations within the novelty dimension (eg: novelties with a wide variation of colors to validate the color dimension). We believe that our work has established a solid groundwork for quantifying the difficulty involved in novelty detection and novelty adaptation. We welcome OWL researchers to employ our difficulty measures as a tool for gaining deeper insights into agent performance.

References

1. Boult, T., Grabowicz, P., Prijatelj, D., Stern, R., Holder, L., Alspector, J., Jafarzadeh, M., Ahmad, T., Dhamija, A., Cli, Cruz, S., Shrivastava, A., Vondrick, C., Scheirer, W.: Towards a unifying framework for formal theories of novelty. In: AAAI (2021)
2. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: A survey. ACM computing surveys (CSUR) **41**(3), 1–58 (2009)
3. Chen, J., Cohn, A.G., Liu, D., Wang, S., Ouyang, J., Yu, Q.: A survey of qualitative spatial representations. The Knowledge Engineering Review **30**(1), 106–136 (2015)
4. Cohn, A.G., Hazarika, S.M.: Qualitative spatial representation and reasoning: An overview. Fundamenta informaticae **46**(1-2), 1–29 (2001)
5. Cohn, A.G., Renz, J.: Chapter 13 qualitative spatial representation and reasoning. In: van Harmelen, F., Lifschitz, V., Porter, B. (eds.) Handbook of Knowledge Representation, Foundations of AI, vol. 3, pp. 551–596. Elsevier (2008)
6. Doctor, K., Task, C., Kildebeck, E., Kejriwal, M., Holder, L., Leong, R.: Toward defining a domain complexity measure across domains. AAAI (2022)
7. Franco, J.P., Yadav, N., Bossaerts, P., Murawski, C.: Where the really hard choices are: A general framework to quantify decision difficulty. bioRxiv (2018)
8. Gamage, C., Pinto, V., Xue, C., Stephenson, M., Zhang, P., Renz, J.: Novelty Generation Framework for AI Agents in Angry Birds Style Physics Games. In: 2021 IEEE Conference of Games, COG 2021 (2021)
9. Gamage, C., Pinto, V., Xue, C., Zhang, P., Nikonova, E., Stephenson, M., Renz, J.: Novphy: A testbed for physical reasoning in open-world environments. arXiv (2023)
10. Gilbert, S., Bird, G., Frith, C., Burgess, P.: Does "task difficulty" explain "task-induced deactivation?". Frontiers in Psychology **3**, 125 (2012)
11. Hernández-Orallo, J., Dowe, D.L.: Measuring universal intelligence: Towards an anytime intelligence test. AI **174**(18), 1508–1539 (2010)
12. Hodge, V., Austin, J.: A survey of outlier detection methodologies. AI review **22**, 85–126 (2004)
13. Kejriwal, M., Thomas, S.: A multi-agent simulator for generating novelty in monopoly. Simulation Modelling Practice and Theory **112** (2021)
14. Langley, P.: Open-world learning for radically autonomous agents. In: AAAI (2020)
15. Li, R., Hua, H., Haslum, P., Renz, J.: Unsupervised Novelty Characterization in Physical Environments Using Qualitative Spatial Relations. In: KR (2021)
16. Martínez-Plumed, F., Hernández-Orallo, J.: Dual indicators to analyze AI benchmarks: Difficulty, discrimination, ability, and generality. ToG **12**(2), 121–131 (2020)
17. Nikonova, E., Xue, C., Pinto, V., Gamage, C., Zhang, P., Renz, J.: Measuring difficulty of novelty reaction. AAAI (2022)
18. Pinto, V., Renz, J., Xue, C., Zhang, P., Doctor, K., Aha, D.W.: Measuring the performance of open-world AI systems. AAAI (2022)
19. Pinto, V., Xue, C., Gamage, C.N., Renz, J.: The difficulty of novelty detection in open-world physical domains: An application to angry birds. arXiv (2021)
20. SAIL-ON-BBA: Science of artificial intelligence and learning for open-world novelty (sail-on) (2019), `https://sam.gov/opp/88fdca99de93ddbb74cd8fb51916ceaa/view`
21. Solaiman, K., Bhargava, B.: Measurement of novelty difficulty in monopoly. In: AAAI (2022)
22. Witten, I., Frank, E., Hall, M.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann Publishers Inc., 3rd edn. (2011)