REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
19-07-2023	Final Report	7/2019 – 7/2023

4. TITLE AND SUBTITLE
Causal Adaptive Decision Aid (CADA)

5a. CONTRACT NUMBER
N00014-19-C-2024

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Aruna Jammalamadaka & Rajan Bhattacharyya

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
HRL Laboratories, LLC
3011 Malibu Canyon Road, Malibu, CA 90265-4797

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Naval Research

10. SPONSOR/MONITOR'S ACRONYM(S)
ONR

11. SPONSORING/MONITORING AGENCY REPORT NUMBER

12. DISTRIBUTION AVAILABILITY STATEMENT
Distribution Statement A: Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPONE NUMBER (*Include area code*)
U	U	U	U		

FINAL REPORT

Causal Adaptive Decision Aid (CADA)

July 19, 2023

Period Covered by the Report: July 29, 2019 to July 28, 2023

Contract Number: N00014-19-C-2024
CLIN#: 0003, CDRL#: A002

Prepared for:
Attn: Jeffrey Morrison, ONR Code: 34
875 North Randolph Street
Arlington, VA 22203-1995
Email: jeffrey.g.morrison@navy.mil

Prepared by:
Dr. Aruna Jammalamadaka and Dr. Rajan Bhattacharyya
HRL Laboratories, LLC
3011 Malibu Canyon Road
Malibu, CA 90265-4797
Email: ajammalamadaka@hrl.com, rbhattac@hrl.com

This material is based upon Work supported by the Office of Naval Research (ONR) under Contract No. N00014-19-C-2024.

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Office of Naval Research (ONR).

Distribution Statement A: Approved for public release; distribution unlimited.

TABLE OF CONTENTS

Section **Page**

Description of All Tasks Performed 1
Accomplishments of the Program 3
2.1 CADASIM Data Generator 3
2.1.1 Methods 3
2.1.2 Evaluation 4
2.2 Symbolic Temporal Reasoning 5
2.2.1 Methods 5
2.2.2 Evaluation 7
2.3 Causal Feature Learning and Display 9
2.3.1 Methods 9
2.3.2 Evaluation 11
2.4 Causal Model Drift Detection and Adaptation 16
2.4.1 Methods 16
2.4.2 Evaluation 17
Recommendations for Future Technology and Research 21
Breakdown of all Contract Costs 22
References 23
Appendices 25
6.1 CADASIM "Red Rover" Scenario and example code 25
6.2 TMPLAR Human Subject Experiment Data 26

LIST OF FIGURES

Figure 1. CADA System Architecture. Symbolic Temporal Reasoner uses domain knowledge to reduce uncertainty in noisy incoming sensor data (purple). Supervised Causal Feature Learning utilizes various types of supervisory input to discover causal decision-making factors from high-dimensional data (blue). Cognitive Adaptation monitors the uncertainty of causal variable representations and relationships to detect anomalies and adapt to potentially non-stationary relationships (green). Given new input data, an optimal Course of Action (COA) is predicted and displayed, along with visually interpretable causal decision-making features (orange). 2

Figure 2. CADASIM training data generation pipeline. 4

Figure 3. An example of CADASIM's interacting weather systems at three different time steps. Thunderstorms exhibit a down flow of wind that radiates outward while cyclones produce wind that rotates away from their "eye." The positions of storms are affected by wind from other storms as well as the presence of any prevailing winds. 4

Figure 4. Example of symbol conversion and axioms flow in STR operation. Raw sensor data is translated to a symbol, after which domain-knowledge-based axioms are applied to reach conclusions about the initial percepts. Code here demonstrates a forward-chaining rule. Following this processing, conclusions may be traced back to the axioms that produced them. 6

Figure 5. The base configuration of NS-CL, which consists of NN-based visual processing elements combined with rule-based symbolic reasoning, driven by semantic VQA curriculum training. 6

Figure 6. Results from data recovery test. Left panel shows the sensor-limited understanding of the CADASIM-generated scene, while the right panel shows the ground truth representation of every entity within the scene. Middle panel is the feature-recovered version of the left sensor-limited version, which clearly marks previous unknown flag affiliations of distant ships, which were inferred through expert rulesets defining enemy ship behavior. One ship (lower right portion of plot) was given an indistinct path outside of STR rulesets, which remained, appropriately, as “Unknown”. 8

Figure 7. Basic operation of the MNIST example running in a neurosymbolic architecture. A NN classifier assigns provisional labels to handwritten digit data, which is then evaluated in a symbolic ruleset (addition) shown on bottom right. The labels are then learned through matching of the symbolic output (sum) to the ground truth sum. 8

Figure 8. With the addition of NovSym (left term) we introduced a conditional symbol that could resolve to either label = 1 or label = 4 probabilistically. The overall accuracy of the neurosymbolic system (gray trace) was reduced slightly, but the conditional symbolic property (orange & blue traces) was discovered accurately. 9

Figure 9. Top: Net_x learns a lower-dimensional embedding X-bar, from which it predicts Y-hat and the bottleneck layer Y-bar of net_y. Similarly, net-y (right) predicts x-hat and x-bar from Y. Bottom: A regularizing loss term is computed between annotated ROIs and heatmaps produced by Grad-CAM [Selvaraju17]. 11

Figure 10 Correlation-based GRAY ship features (left) show higher importance near BLUE’s shortest-path course (blue horizontal line), indicating that they should impact BLUE’s navigational decision. After active causal feature learning, added examples increase

the entropy (and thereby decrease the importance) of GRAY ships in the same area (yellow circle). 12
Figure 11. Two estimated clusters of P(Y|X) provide a division in utopian point score corresponding to different decision-making styles. 13
Figure 12. Averaged eye-tracking data per cluster and DSS level. Participants in the first cluster (top row) and the second cluster (bottom row) both tend to focus only on the route table in the lower half of the screen....14
Figure 13. El Niño Dataset [Chalupka16]. Left: At the macro-level, westerly equatorial winds are a known cause of El Niño, characterized by deviations in sea surface temperature within the Niño 3.4 region (120◦W-170◦W, 5◦N-5◦S) (yellow). Right: Four examples of microvariables X and Y, corresponding to zonal WS and SST maps from a region of the Equatorial Pacific Ocean. 14
Figure 14. Visualization of 4 concepts discovered by CAE (Top) and 2 concepts discovered by SCAE (Bottom) for the El Niño dataset. Gray boxes indicate coarsening of correlational concepts into causal ones. SCAE results are also labeled using [Chalupka16] discovered categories for WS and SST: Easterly Equatorial (EEqt), Westerly Equatorial (WEqt), Easterly North of Equator (EN) and Easterly South of Equator (ES), Cold, El Niño, La Niña, and Warm. 15
Figure 15. Synthetic dataset, adapted from [Höltgen21]. Left: Generative model. Right: Four examples each of X and Y, which are 8x8 gray-scale images. The causal effect of interest corresponds to y2 (yellow), which also serves as ROI supervision. 15
Figure 16. Grad-CAM results from partial supervision. Each of the dashed vertical lines indicate a different proportion of image masks provided to the system, according to the percentage below. Greater than 50% supervision is needed in order to collapse the four correlated macrovariables for netX and netY to the two known causal macrovariables for this synthetically generated dataset....16
Figure 17. Pendulum simulation and corresponding causal graph....17
Figure 18. Best F1 scores for CausAnom, and the baselines: auto-encoder, and auto-encoder (with effect vars only). Case 1 models the standard training scenario and Case 2 models the biased training data scenario. 18
Figure 19. Nominal (left) and Off-nominal (right) causal graphs for the Ship Power dataset. 19
Figure 20. Area under MSE (R-AUC) for CausAnom and MLP ensemble baseline. This metric estimates how well the model uncertainty matches the prediction accuracy in experiments on the Ship Power dataset. 20
Figure 21. Example scenario involving RED and BLUE ships/agents. Starting from the top-left frame and moving toward the bottom-right, the BLUE ship is tasked with moving to a position far in the east. As soon as BLUE enters the security perimeter guarded by the RED ship, RED begins to intercept BLUE. Once the BLUE ship determines that they are being intercepted by RED, they take evasive maneuvers, ultimately leading them to backtrack out of the security perimeter. Once BLUE is outside the security perimeter, RED begins returning to the center of their area. But, realizing that they are no longer being threatened, BLUE resumes their course eastward, again crossing into the security area. Immediately, RED returns to their pursuit of BLUE, but this time, they are behind BLUE, enabling BLUE to travel eastward through (and then out of) the security area. 25
Figure 22. Code snippet showcasing ease of naval scenario creation in CADASIM. 26

Description of All Tasks Performed

Logistics and planning personnel are overloaded by the increasing number of high-dimensional data layers they are expected to analyze in short amounts of time (e.g., a 24-hr battle rhythm). The goal of the Causal Adaptive Decision Aid (CADA) is to facilitate the foraging and sensemaking steps of their analyses by rapidly summarizing these layers and displaying the cause-and-effect concepts which are relevant to recommend explainable courses of action (COAs). A causal (as opposed to correlation-based) model for COA recommendation has the benefits of being more robust to novel contexts by mitigating sample bias issues, while also playing a key role in analytic tradecraft. We achieve these goals by developing and delivering software modules related to the following four major tasks:

- **CADASIM Data Generator:** Simulates realistic naval scenarios, from noisy sensor data to "human" decision, to be used as a benchmark data set for subsequent tasks.
- **Symbolic Temporal Reasoning:** Leverages domain expertise in the form of (neuro-)symbolic rules and axioms to reduce uncertainty in naval scenarios by inferring unknown attributes.
- **Causal Feature Learning and Display:** Discovers cause-and-effect concepts from high-dimensional datasets where interventions are not possible, but some form of domain expert input may be available.
- **Causal Model Drift Detection and Adaptation:** Monitors uncertainty of data-driven causal models and concepts and differentiates between data and concept drift, thereby minimizing the need for model retraining.

The end-to-end system has been filed as a patent, and some aspects of the causal feature learning methods and results have been published as a paper ([Jammalamadaka23]). Although we were unable to test the end-to-end system due to lack of benchmark datasets needed to showcase all aspects, the tasks outlined above can conceptually fit together as integrated software modules as shown in Figure 1.

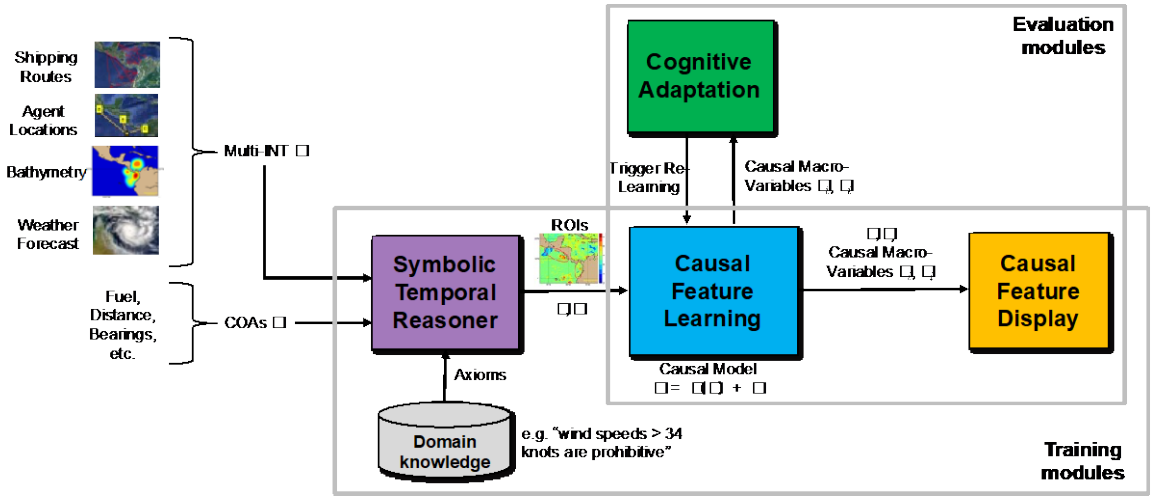

Figure 1. CADA System Architecture. Symbolic Temporal Reasoner uses domain knowledge to reduce uncertainty in noisy incoming sensor data (purple). Supervised Causal Feature Learning utilizes various types of supervisory input to discover causal decision-making factors from high-dimensional data (blue). Cognitive Adaptation monitors the uncertainty of causal variable representations and relationships to detect anomalies and adapt to potentially non-stationary relationships (green). Given new input data, an optimal Course of Action (COA) is predicted and displayed, along with visually interpretable causal decision-making features (orange).

Accomplishments of the Program

2.1 CADASIM Data Generator

- **Goal:** Simulate realistic naval scenarios, from noisy sensor data to "human" decision, to be used as a benchmark data set for subsequent tasks.
- **Accomplishments:** Developed CADASIM Python 3 package, which surpasses existing open-source naval simulators (e.g., UTSeaSim [Agmon11]) by enabling a novice programmer to easily:
    - Create complex and dynamic environments which provide agents (e.g., ships) with situational awareness that may be noisy, incomplete, and involve asymmetric sensing abilities.
    - Specify long-term goals (missions), mobility patterns, and utility functions for a variety of agent types.
    - Customize sensors with state-based noise functions and failure modes, fading characteristics, transmission power, and a noise floor for individual sensors.

2.1.1 Methods

CADASIM is implemented in Python 3 and has a modular and functional architecture that enables the creation of simulation scenarios in a customizable way. This includes the ability to customize sensors with state-based noise functions and failure modes so that (for example) a wind gauge can be less precise when winds are exceedingly high; as well as specify fading characteristics, transmission power, and a noise floor for individual radar sensors to instantiate scenarios with agents (ships) have asymmetric sensing abilities.

The environment can contain winds and weather systems that dynamically interact with one another and effect the navigation abilities of the ships. CADASIM implements a prevailing wind model (e.g., trade winds), simple thunderstorms and rotational storms (e.g., tropical storms); see Figure 3. Storm sizes and strengths can be dynamic so that they can grow and shrink over time. All wind models can be made gusty and can change their directionality. Currently, winds only effect the trajectory of ships, requiring them to make corrections to stay on course. Ocean surface conditions (waves) that are influenced by the wind and in turn influence ships' max safe speed are not yet implemented.

CADASIM scenarios, defined by initial conditions and low-level agent mobility, can be user-defined, or randomly generated. The simulator runs every COA for every scenario and records performance metrics, e.g., mission success, time taken, risk incurred (based on distance to enemy ships). To save time and efficiently generate alternate "views" of the simulated data due to varying operational phases or decision-making styles, user-specified utility functions are used to determine optimal COAs. The utility functions are applied as a post-processing step after the simulator performs the computationally expensive task of computing and measuring the outcomes of the different actions, allowing us to filter the data set down to appropriate training sets for specific causal decision-making factors (here the "effect" of interest is assumed to be the COA itself). The scenario-COA combination with the highest utility is added to the data set. This process is outlined in Figure 2.

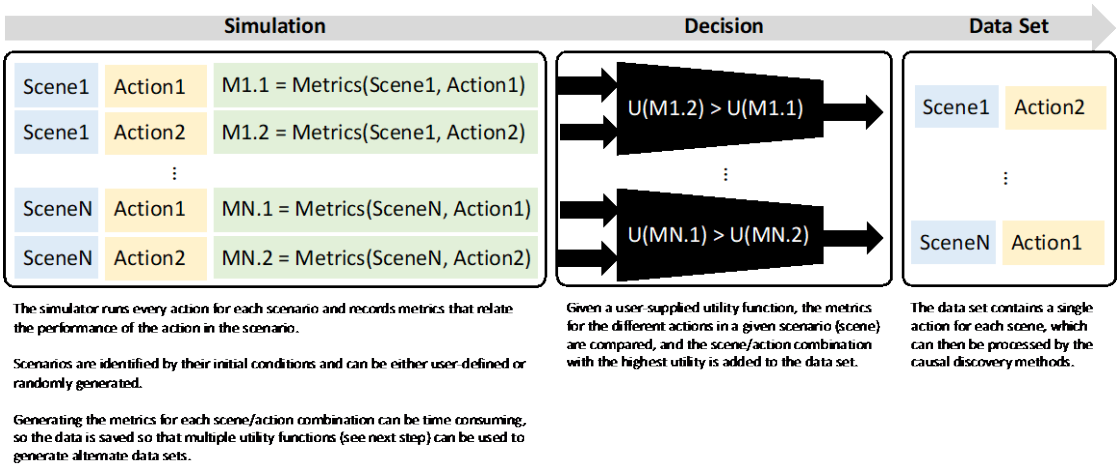

Figure 2. CADASIM training data generation pipeline.

2.1.2 Evaluation

CADASIM's realistic noise sets and environment-dependent agent movements were guided and evaluated informally through discussions with Naval subject matter expert (SME) Glenn White, who believed at that time that our design choices make it easily integrable with existing Naval simulators like Joint Semi-Automated Forces (JSAF). Discussions with Harold Hawkins regarding making the simulation package open source and available to other Science of AI performers in the unclassified space were inconclusive, possibly because both Glenn White and Harold Hawkins retired in that timeframe. Figure 3 shows example simulated output for METOC information (weather layer of multi-INT data). An example scenario involving red and blue agents along with a code snippet for scenario generation can be found in Appendix 6.1.

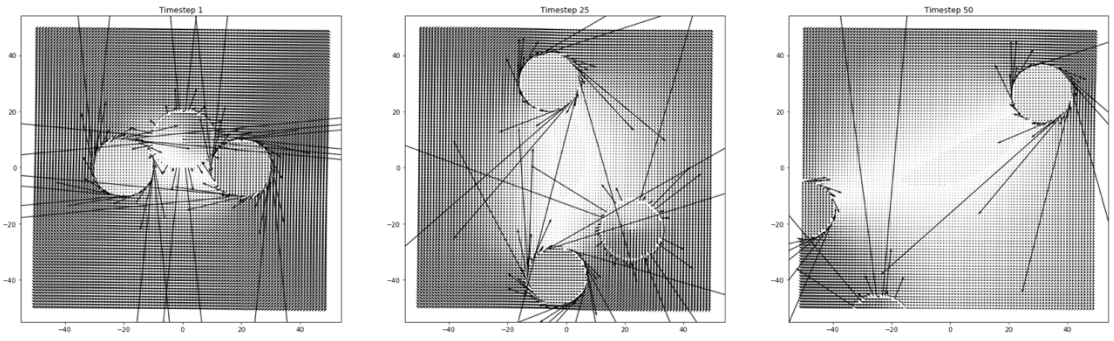

Figure 3. An example of CADASIM's interacting weather systems at three different time steps. Thunderstorms exhibit a down flow of wind that radiates outward while cyclones produce wind that rotates away from their "eye." The positions of storms are affected by wind from other storms as well as the presence of any prevailing winds.

2.2 Symbolic Temporal Reasoning

- **Goal:** Leverage domain expertise in the form of (neuro-)symbolic rules and axioms to reduce uncertainty in naval scenarios by inferring unknown attributes.
- **Accomplishments:** Developed probabilistic symbolic reasoning capabilities in Python 3 which enable proof-of-concept demonstrations for:
    - Inferring critical uncertain and unknown decision-making factors in a realistic naval scenario generated by CADASIM.
    - Assigning and inferring characteristics of unknown neuro-symbolic symbols as probabilistic combinations of known symbols.

2.2.1 Methods

STR is a form of symbolic reasoning, in which (primarily) axiom-based logic [Landy14] is employed to make conclusions about the past or future based on a set of observables obtained in the present [Vila94]. In CADA, this can be used to draw critical conclusions about the current situation, such as the likely disposition of detected ships, areas of uncertainty in the geography, and other critical items that rely on prior experience or domain expertise. Raw "sensor" data (e.g., from the BLUE ship in CADASIM) is used to generate semantically rich symbols, which are then evaluated over our domain-knowledge-derived axioms. These axioms determine whether the criteria for pre-defined past (abduced) and future (deduced) scenarios are met and produce a set of conclusions that apply to the present set of "givens" provided by the sensors (Figure 4).

One of the advantages of employing this type of system within CADA is that it provides features to causal discovery that would otherwise be unavailable; for example, raw sensor information may provide little information about the disposition of an unknown contact, whereas a skilled observer may be able to infer this information through prior intelligence of expected contacts in the area or referring to a library of known contact behaviors. In this way, it can also help reduce uncertainty due to sensor noise, by inferring missing data using information present before or after the current time point. The other advantage lies in plausibility; utilizing axiomatic logic can provide additional information from a realistic situation, but can also flag conflicts in unrealistic ones by identifying axiomatic "collisions" leading to contradictory conclusions [Besold17]. Such situations can occur in the process of active causal feature learning (Section 2.3.1.1), in which scenario attributes that are intervened upon may result in impossible outcomes (e.g., a ship is determined to be both hostile and friendly).

2.2.1.1 Symbolic Temporal Reasoner

In a practical scenario, rulesets which encode domain expert knowledge may be intelligence data or compiled knowledge of naval strategies; in our simplified case, the domain dealt with ship behaviors based on disposition (e.g., "RED" and "GRAY" ships representing hostile and neutral ships with respective behaviors) and simple physics (e.g., ships may not teleport). To evaluate this domain knowledge and make inferences, we developed a logic programming approach to operate on CADASIM data, based on the Python Knowledge Engine (PyKE, http://pyke.sourceforge.net). These inferences are then

compiled into an output symbol set, which include additional pieces of information (e.g., symbol assignments/attributes) about that are recovered through the expert knowledge encoded in the rulesets. A deterministic script then re-populates the raw coordinate data with this recovered data to produce a more complete picture of the sensor-limited data. That more complete picture is then fed into the causal feature learning modules.

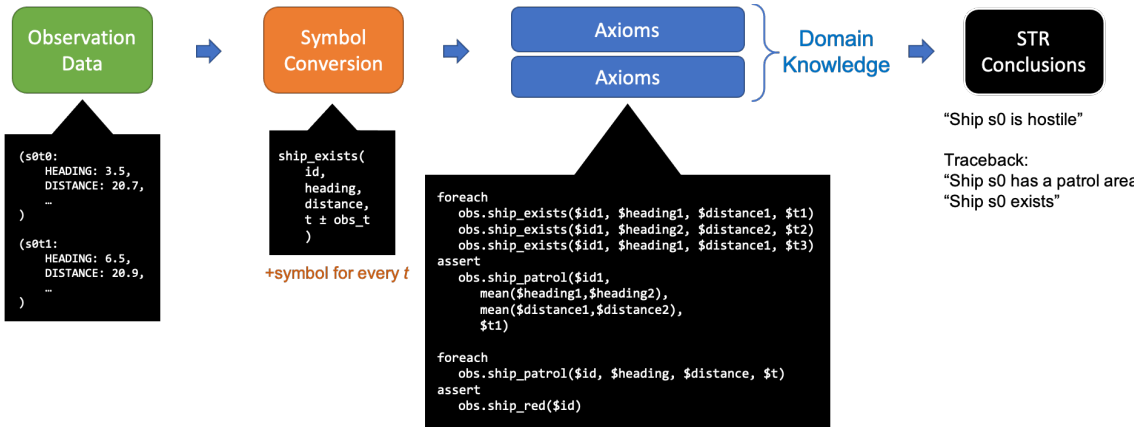

Figure 4. Example of symbol conversion and axioms flow in STR operation. Raw sensor data is translated to a symbol, after which domain-knowledge-based axioms are applied to reach conclusions about the initial percepts. Code here demonstrates a forward-chaining rule. Following this processing, conclusions may be traced back to the axioms that produced them.

2.2.1.2 Neuro-symbolic Temporal Reasoner

A limitation of the above method is the necessity for manually created and curated symbols and rulesets. To reason about unknown symbols, we leverage neuro-symbolic learning and reasoning. Neuro-symbolic learning combines the raw-data processing abilities of artificial neural networks (ANNs) with the traceable, higher-order reasoning abilities of logic-based rules to produce a hybrid system capable of both. The ANNs form the sub-symbolic symbol grounding, which can then be run as symbolic executions via rule-based processing. Our method is based on the Neurosymbolic Concept Learner (NS-CL) [Mao19] which utilizes a Visual Question Answering (VQA) style curriculum learning, which we have modified to handle probabilistic or provisional data. The architecture is like that of the NS-CL, utilizing a pre-trained Mask R-CNN [He17] to

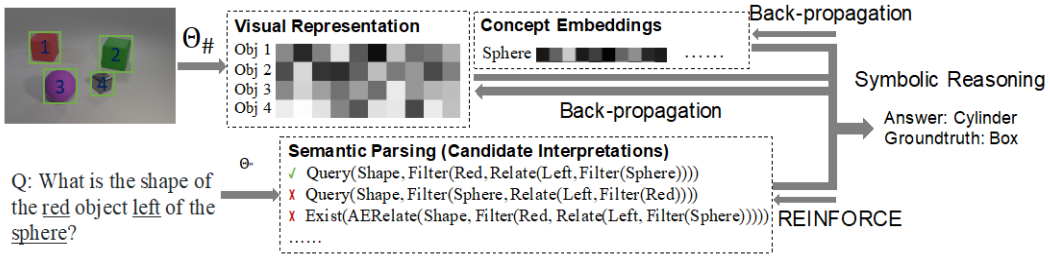

Figure 5. The base configuration of NS-CL, which consists of NN-based visual processing elements combined with rule-based symbolic reasoning, driven by semantic VQA curriculum training.

generate object proposals in a visual scene, which are then sent to ResNet-34 [He15] to extract region- and image-based features of potential objects. The features extracted are concatenated into a vector representation of objects and their respective features and embedded into a visual-semantic space, from which object attributes can be classified and quantized based on cosine distances computed from known object attributes (Figure 5). VQA-style queries can then be utilized to first train individual attributes, which through a guided curriculum grow more complex as more object/attribute concepts are learned by the system. After this training step, the system is configured to accurately report on highly generalized visual fields containing complex objects [Mao19]. To increase generalizability and form the basis for probabilistic reasoning, we take advantage of NS-CL's inherent error-tracing capabilities to generate provisional symbols with missing, novel, or improperly formatted data.

2.2.2 Evaluation

The following results are small proof-of-concept demonstrations of the utility of STR in processing and augmenting scenarios with ambiguous sensor readings, which is of high relevance as the realism and practicality of test scenarios increases. They serve as a critical demonstration of symbolic processing concepts that may be rolled into strategies to learn novel versions of both symbols and rulesets through training on generalized examples of naval scenarios, using a hybrid approach of symbolic processing and machine learning.

2.2.2.1 Symbolic Temporal Reasoner Result

Using CADASIM, a scenario was generated using a lower-powered radar sensor mounted to the ego (blue) ship that prevented accurate accounting of ship affiliations and ship detection at long ranges. This resulted in an incomplete scenario shown in the left panel in Figure 6, which correctly identifies the affiliation of nearby ships and marks unknown ships appropriately with an ambiguous marker. A blue area map was added post-data generation as an indication of radar range. The panel at right shows the ground-truth generation of this world scenario wherein all entities in the world are represented accurately as assigned at simulation start. As designed, the "sensor-limited" version allows accurate detection out to a certain range, then begins to falter first in terms of flag affiliation, then fails to detect ships altogether at long ranges. A short time domain is also simulated to provide some other information about the ambiguous radar contacts; here, we took advantage of the simulation properties to define certain ship behaviors (e.g., patrolling) to flag affiliations. This represents the type of expert knowledge that could be exercised to understand additional details about the scene. The middle panel shows a recovered features version of the right panel "sensor-limited" scene, which shows the recovery of some flag affiliations of distant ships due to STR rulesets that utilize ship behaviors to identify ship color. Though not easily visualized, the STR also "stores" the identification of red ships. This ensures continuity and the ability to reason about object

permanence, even if the radar contact should navigate away from detection range later in the scenario.

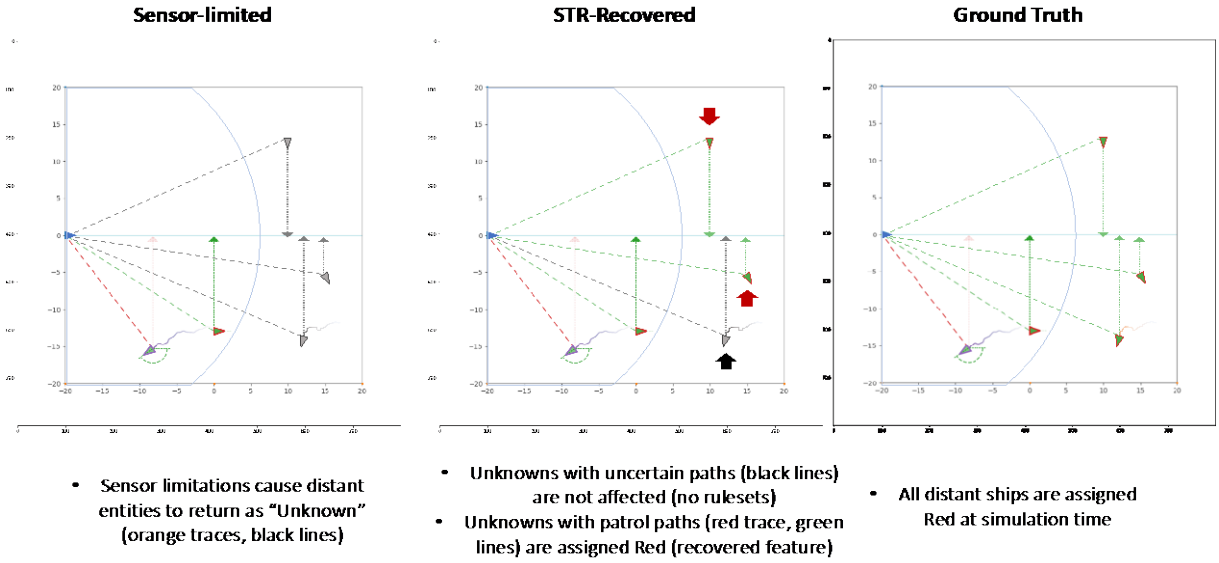

Figure 6. Results from data recovery test. Left panel shows the sensor-limited understanding of the CADASIM-generated scene, while the right panel shows the ground truth representation of every entity within the scene. Middle panel is the feature-recovered version of the left sensor-limited version, which clearly marks previous unknown flag affiliations of distant ships, which were inferred through expert rulesets defining enemy ship behavior. One ship (lower right portion of plot) was given an indistinct path outside of STR rulesets, which remained, appropriately, as "Unknown".

2.2.2.2 Neuro-symbolic Temporal Reasoner Result

Our prototype neurosymbolic system, based on the library DeepProbLog (Deep, Probabilistic, Logic) [Manhaeve18] utilizes the MNIST handwriting dataset to test the ability of neurosymbolic sensemaking in the context of (symbolic) arithmetic operations on optical character recognition (OCR) data (Figure 7). In addition to the example MNIST dataset consisting of digits 0-9, we introduced a novel visual character (NovSym) with conditional, probabilistic properties (Figure 8). As expected, a purely neural classifier failed to detect the conditional aspect of NovSym, which resulted in a slight decrease in classification accuracy but no indication of the conditional nature of the symbol (Figure 7). We then expanded our investigation to more complex datasets with the intention of

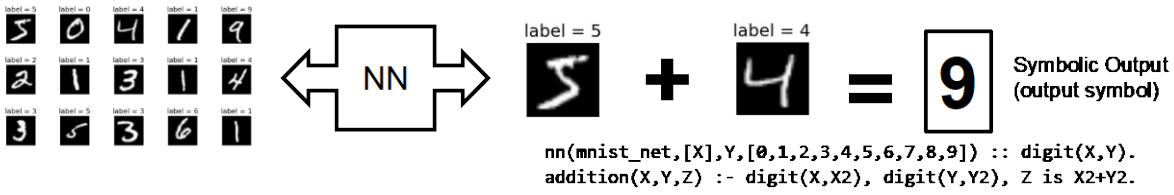

Figure 7. Basic operation of the MNIST example running in a neurosymbolic architecture. A NN classifier assigns provisional labels to handwritten digit data, which is then evaluated in a symbolic ruleset (addition) shown on bottom right. The labels are then learned through matching of the symbolic output (sum) to the ground truth sum.

building in capabilities to handle such novel symbols in a sensemaking task of greater complexity (e.g., CLEVR Visual Question Answering (VQA) dataset [Johnson17]).

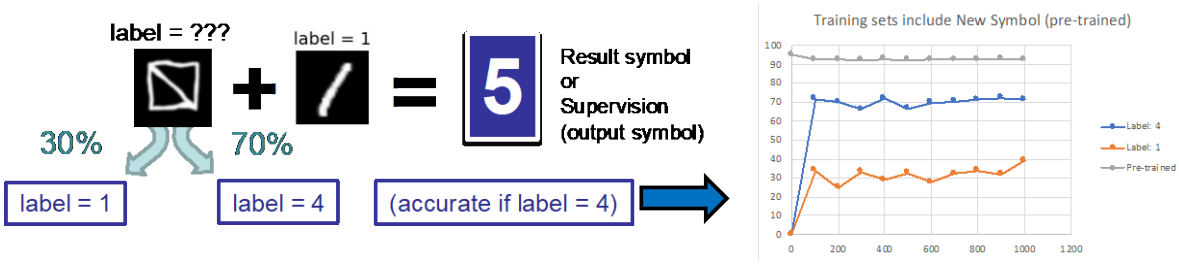

Figure 8. With the addition of NovSym (left term) we introduced a conditional symbol that could resolve to either label = 1 or label = 4 probabilistically. The overall accuracy of the neurosymbolic system (gray trace) was reduced slightly, but the conditional symbolic property (orange & blue traces) was discovered accurately.

2.3 Causal Feature Learning and Display

- **Goal:** Discover cause-and-effect concepts from high-dimensional datasets where interventions are not possible, but some form of domain expert input may be available.
- **Accomplishments:** Developed Python 3 software packages, which include
  - Demonstration of causal feature learning using CADASIM Red Rover scenario.
  - Discovery of decision-making factors in the 2019 TMPLAR Decision Support System (DSS) data
  - Causal Autoencoder framework and resulting improvements to El Niño weather phenomena discovery (details published in [Jammalamadaka23]).

2.3.1 Methods

Assuming a historically observed dataset of high-dimensional paired observations {X,Y} wherein we believe that aspects of X cause aspects of Y, causal feature learning aims to extract lower-dimensional concepts (also called features or factors in this report) which preserve all cause-effect relationships within the data. Causal models are well-known as the "holy grail" of machine learning, in that they should be more robust than correlation-based models to commonly occurring reasons for drops in operational performance like training data sample bias and other environmental changes.

The hard challenge is that causal theory dictates that causality cannot be determined without the ability to intervene on the data generating process of the data, which is not possible in many observed datasets due to ethical, physical, or cost-based constraints. In [Chalupka15] they prove the *Causal Coarsening Theorem*, which dictates that causal concepts can be derived from further merging (or coarsening) the correlation-based concepts according to the effects (outcomes) of targeted interventions on specific attributes of the causes. In CADA, we have explored several ways to obtain the correlation-based concepts from observed historical data, and to coarsen them to causal ones. These are detailed below.

2.3.1.1 Causal Feature Learning (CFL)

[Chalupka15] attempts to solve this problem using a method called Causal Feature Learning (CFL). The method finds equivalence classes of the conditional distribution P(Y|X) by clustering together X's that are observed to correlate with similar Y's and Y's which are observed to correlate with similar X's. This grouping produces a *correlational* partition of the space of X and Y such that every new observation falls into a partition cell or cluster which corresponds to a categorical concept value. As mentioned above, they prove that the *causal* concepts and partition resulting from equivalence classes the interventional distribution "P(Y|*do*(X=x))" can be found by either keeping or further merging of these correlational partition cells/clusters according to a minimal set of interventions described next.

Causal Coarsening via Active Causal Feature Learning

This method, like the "causal manipulator" proposed in [Chalupka15], identifies observations which are closest to partition boundaries to construct synthetic X observations using targeted interventions which, when fed to an "oracle" (e.g., an expert decision-maker or the CADASIM ground truth simulator), provides a new Y, allowing us to update and refine the correlational partition to the ground truth causal partition.

Causal Coarsening via Information-Theoretic Causal Scoring

In cases where we may not have access to decision-maker feedback or other forms of supervision, we explored several recent information-theoretic "causal scoring methods" [Guyon19] to determine cause-and-effect concepts. These methods were adapted in CADA to score potential causal coarsenings and arrive at an estimated causal partition. We found the minimum description length method in [Marx18] to work on simple synthetically generated data, but to be unsuccessful in providing reasonable results on more complex navy-relevant data, where underlying concepts could potentially be correlated. The lesson learned here is that some form of human supervision is far better than using information-theoretic methods in terms of obtaining causality from correlation. Since this succinctly summarizes the results of our evaluations, we do not provide additional results in the next section.

2.3.1.2 Supervised Causal Autoencoder (SCAE)

The drawbacks of CFL are that it does not allow for continuous-valued concepts (which provide a richer lower-dimensional representation), does not provide an integrated method for determining the number of causal macrovariables (we had to use various standard post-hoc cluster metrics), and does not lend itself easily to region of interest (ROI)-based supervision. To overcome these drawbacks, we extend a neural architecture called the Causal Autoencoder [Höltgen21] by adding a regularization term which forces the encoders that map the high-dimensional X (or Y) to the lower-dimensional causal features X-bar (or Y-bar) to pay attention to specific ROIs within the input (Figure 9). The resulting lower-dimensional representations X-bar and Y-bar are continuous-valued vectors comprised of multiple concepts, as opposed to the categorical concept values obtained from CFL.

Causal Coarsening via Region of Interest-based regularization

This method to obtain causal concepts incorporates subject matter expertise provided as visual ROIs (e.g., image masks or eye-tracking heatmaps). That input is compared to the Grad-CAM heatmap [Selvaraju17] of either or both encoders to ensure learned causal concepts preserve information in those regions (Figure 9). For more detail please see [Jammalamadaka23].

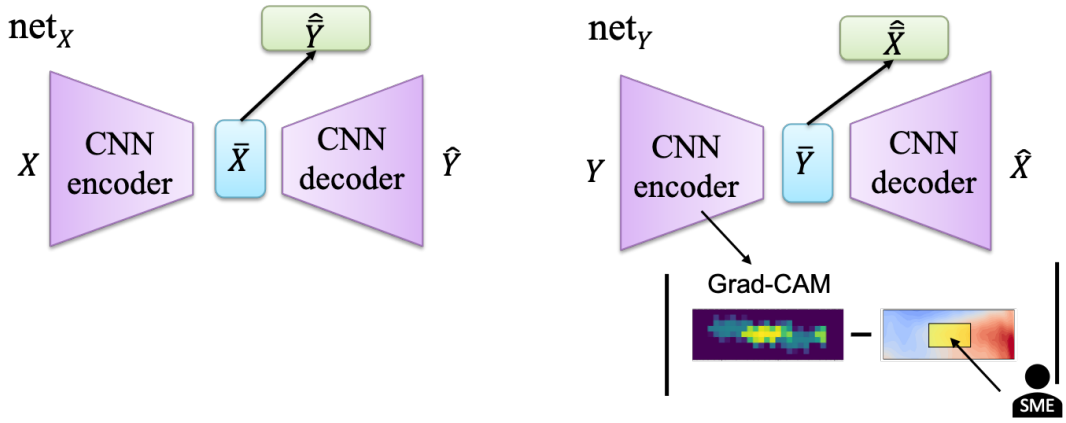

Figure 9. Top: Net_x learns a lower-dimensional embedding X-bar, from which it predicts Y-hat and the bottleneck layer Y-bar of net_y. Similarly, net-y (right) predicts x-hat and x-bar from Y. Bottom: A regularizing loss term is computed between annotated ROIs and heatmaps produced by Grad-CAM [Selvaraju17].

2.3.2 Evaluation

During this program, we have tested our causal feature learning methods on numerous synthetic and real-world datasets, a subset of those results is described in this section. We stopped evaluating our methods on data from the CADASIM simulator in year two of the program, in lieu of using real-world human subject and climate science data. The primary reason for this is that Glenn White was no longer available to vet our simulation design decisions and without his guidance we had concerns about generating overly contrived scenarios to showcase our causal feature learning methods. We were also eager to prove our methods on real observed data to enhance our ability to transition to real naval route-selection and planning problems. The trade-off in using real-world data is that synthetic data has known causal ground truth that can be used to measure the accuracy of our results, whereas real data requires some type of expert verification.

2.3.2.1 CFL Results

CADASIM Red Rover Results

To test the benefits of causal vs correlational models we utilize the Red Rover scenario described in Appendix 6.1. The X variable consists of multi-INT layers of ship locations and weather patterns. The Y variable consists of the self ("BLUE") ship's COA. The goal of causal feature learning is to determine that the neutral ("GRAY") ship's location correlates with that of the adversarial ("RED") ship but is not a causal factor for selection of BLUE's optimal COA. Causal coarsening is performed via active causal feature learning, using CADASIM as an oracle. CADASIM generates visualizations of paths taken by each ship type during the observation period, in addition to logging numeric feature data. Figure 10 displays the causal features for in the images pertaining to the GRAY ship. Heatmaps should be high entropy (uninformative) since GRAY ship corresponds to confounding factor, not causal of BLUE's optimal COA.

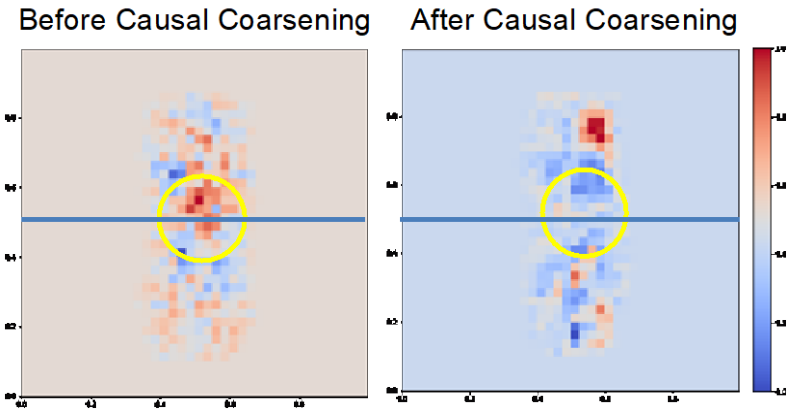

Figure 10 Correlation-based GRAY ship features (left) show higher importance near BLUE's shortest-path course (blue horizontal line), indicating that they should impact BLUE's navigational decision. After active causal feature learning, added examples increase the entropy (and thereby decrease the importance) of GRAY ships in the same area (yellow circle).

TMPLAR DSS Results

We applied CFL on real naval decision-making data collected in 2019 for the Human-AI Symbiosis Science of AI project, and kindly provided by David Sidoti at NRL-Monterey, Dr. Krishna Pattipati at University of Connecticut, and Dr. Mollie McGuire at Naval Postgraduate school. The experiment was performed to evaluate the effectiveness of three decision support systems (DSS) containing varying amounts of information from the TMPLAR automated tool for surface vessel route planning. Each DSS aims to facilitate the selection of situationally appropriate and tactically sound routes by Naval Postgraduate school participants. The dataset contains 1827 raw data samples and 77 features for 29 participants after filtering for missing fields. Please see Appendix 6.2 for a detailed description of each feature, and [Uziel20] for more details on the associated eye-tracking data and DSS frameworks.

The high-dimensional X variable consists of numeric route features (e.g., fuel, distance, bearings, wind speeds) normalized relative to the other route options offered for that scenario. We choose the Y variable to be the utopian point score of the selected route

(COA), which is an indication of the quality (lower is better) in multi-objective optimizations where there is no other clear definition of which selection is the "best". The goal of CFL in this case is to determine the specific route features and value ranges which cause a better or worse utopian score.

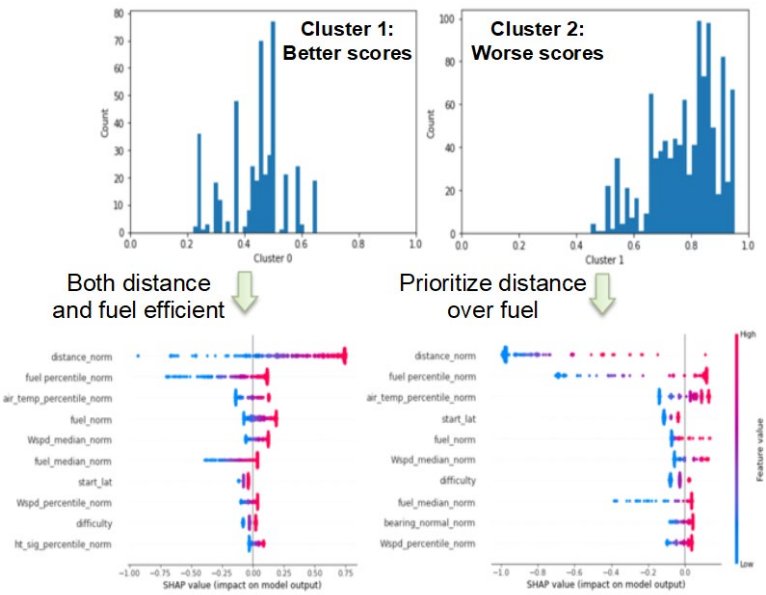

Figure 11. Two estimated clusters of P(Y|X) provide a division in utopian point score corresponding to different decision-making styles.

CFL without the coarsening step was used to extract correlation-based feature importances, by first training a neural network with input X and output Y, and then clustering our estimate of P(Y|X) which comes from the second to last layer of the architecture. This process produces two stable clusters which we hypothesize to correspond to a division in decision-making styles of participants. To visualize what the important features were for each decision-making style, we utilize SHapley Additive exPlanations (SHAP) [Lundberg20], a game theoretic approach to extract feature importances from trained neural networks. The results in Figure 11 show that Cluster 1 participants focused on minimizing distance and fuel and therefore have higher utopian point scores, whereas Cluster 2 participants tend to favor lower distance at the cost of higher fuel, leading to worse utopian point scores (since the utopian point in this case is the minimum distance and minimum fuel across all routes).

Our next step was to determine whether eye-tracking data from the experiment showed a similar division in decision-making style. If so, it could potentially be used as ROI-based supervision to train novice planners on what aspects of the DSS they should focus on. However, as shown in Figure 12, we found that all participants (regardless of utopian score, cluster assignment, or DSS type) primarily focused on the route table in the UI. These results agree with analysis of mouse-click data by Matthew Macesker (UConn)

which also confirmed that there was very little variation across subjects in aspects of the DSS that were used.

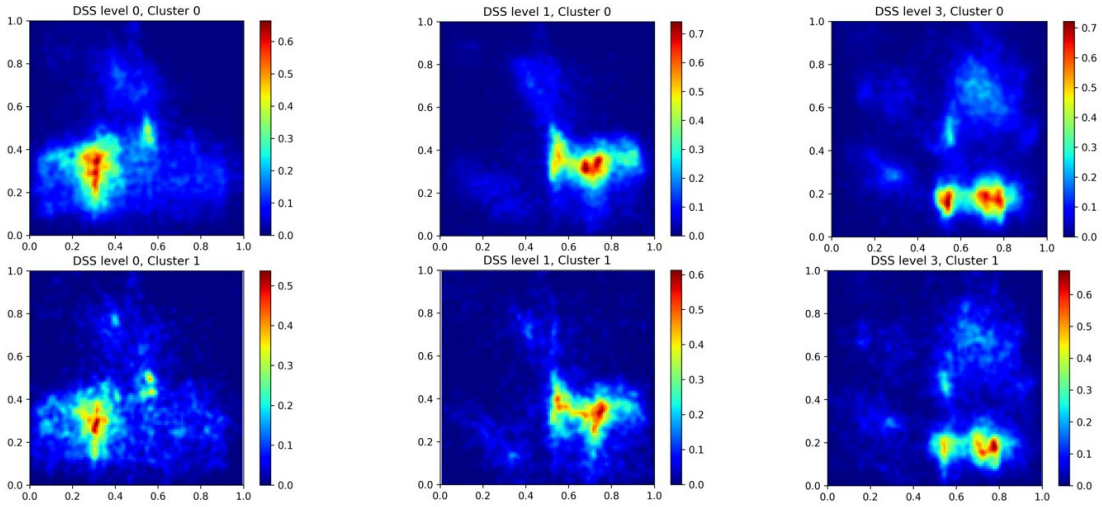

Figure 12. Averaged eye-tracking data per cluster and DSS level. Participants in the first cluster (top row) and the second cluster (bottom row) both tend to focus only on the route table in the lower half of the screen.

2.3.2.2 SCAE Results

We evaluate our method with two experimental datasets: the synthetic dataset (with causal ground-truth) introduced in [Höltgen21], and the climate dataset introduced in [Chalupka16]. In both datasets, we empirically show that given enough supervision, our supervised framework can perform the aforementioned ROI-based causal coarsening. For more detail than what is described below, please refer to [Jammalamadaka23].

El Niño Dataset Results

For this experiment we leverage the real-world climate science dataset introduced in [Chalupka16], where aspects of high-dimensional zonal wind speed maps ("WS", X) cause aspects of sea surface temperature maps ("SST", Y) within the same region of the Equatorial Pacific Ocean. The goal of SCAE is to discover the El Niño phenomenon (concept) from 35 years of climate data and verify that the causal concept associated with this effect concept corresponds to a larger-than-average westerly equatorial wind (Figure 13).

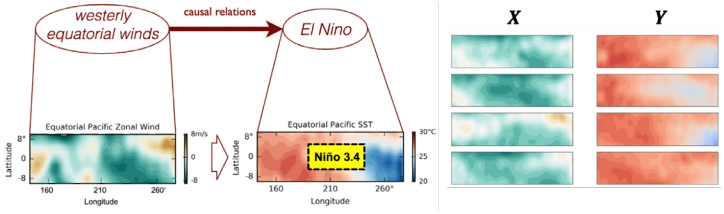

Figure 13. El Niño Dataset [Chalupka16]. Left: At the macro-level, westerly equatorial winds are a known cause of El Niño, characterized by deviations in sea surface temperature within the Niño 3.4 region (120◦W-170◦W, 5◦N-5◦S) (yellow). Right: Four examples of microvariables X and Y, corresponding to zonal WS and SST maps from a region of the Equatorial Pacific Ocean.

Figure 14 shows that we are able to validate our results according to the known cause-and-effect concept dynamics described above. We can also visually validate that the addition of supervision coarsens the correlation-based concepts, according to theory. As a third method of validation, we also show a 15% improvement in the precision of the discovered El Niño concept compared to the CFL results in [Chalupka16], where the metric is introduced. Finally, due to the coarsening our method provides, SCAE uses 2 instead of 4 concepts discovered by CAE to model the same known causal dynamics. This should provide a more succinct summary for analysts using this method and therefore lower their cognitive load.

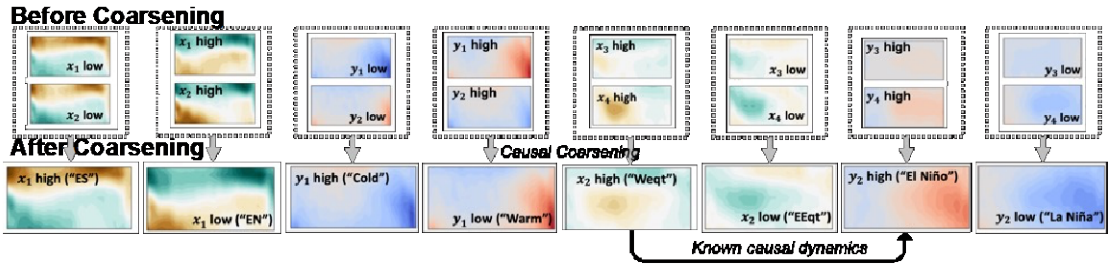

Figure 14. Visualization of 4 concepts discovered by CAE (Top) and 2 concepts discovered by SCAE (Bottom) for the El Niño dataset. Gray boxes indicate coarsening of correlational concepts into causal ones. SCAE results are also labeled using [Chalupka16] discovered categories for WS and SST: Easterly Equatorial (EEqt), Westerly Equatorial (WEqt), Easterly North of Equator (EN) and Easterly South of Equator (ES), Cold, El Niño, La Niña, and Warm.

Partial Supervision Results on Synthetic Data

Subject matter expertise, even in the form of highlighted image regions, is often expensive and hard to obtain. To test how much supervision is needed to coarsen correlation-based concepts to causal ones, we vary the percentage of image masks provided for a random selection of images. We utilize a synthetically generated dataset for this test since the known ground truth enables more accurate evaluation of the results. In this dataset, X and Y are random variables in R^64 which can be visualized as 8x8 pixel gray-scale images. The generative model for the data has two-dimensional macrovariables: X-bar = (x1, x2) and Y-bar = (y1, y2) which correspond to pixel value averages in the top/bottom halves of X and the right/left halves of Y, respectively. In the data generating process, x1 and y1 have a common cause c1 (i.e., they are correlated but x1 does not directly cause y1), whereas x2 is a direct cause of y2. ROI supervision is

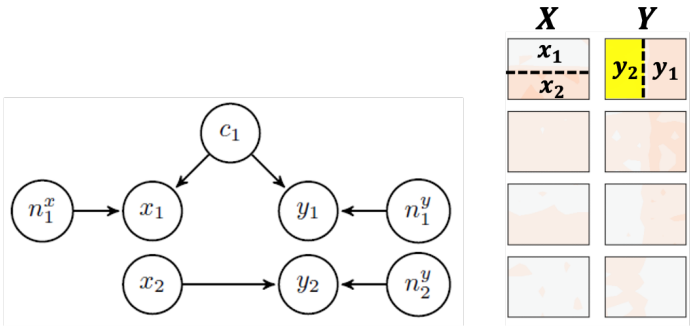

Figure 15. Synthetic dataset, adapted from [Höltgen21]. Left: Generative model. Right: Four examples each of X and Y, which are 8x8 gray-scale images. The causal effect of interest corresponds to y2 (yellow), which also serves as ROI supervision.

imposed by highlighting the left half of Y images corresponding to y2 (Figure 15) indicating that this is an effect of interest.

Resulting Grad-CAM [Selvaraju17] heatmaps in Figure 16 show that the number of informative neurons drops from four (correlated) to two (causal) after 50% of images are supervised. Although this may seem like a lot of images to annotate depending on the size of the dataset, it is important to note that if the discovered concepts are truly causal in nature, they should be robust to added observation, i.e., the number of necessary annotations should stay constant as dataset size grows.

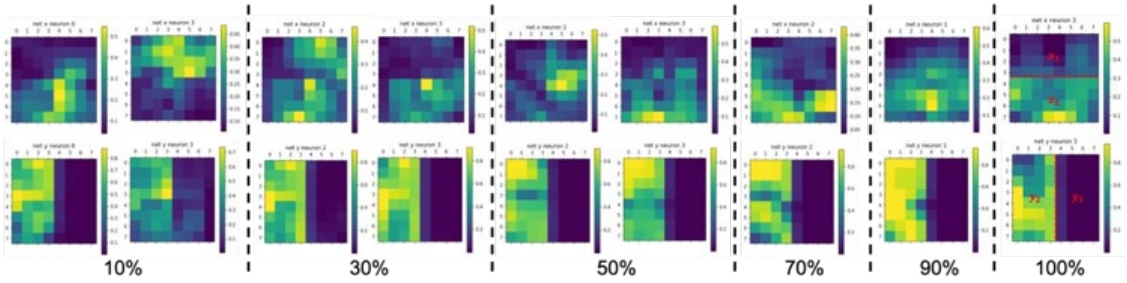

Figure 16. Grad-CAM results from partial supervision. Each of the dashed vertical lines indicate a different proportion of image masks provided to the system, according to the percentage below. Greater than 50% supervision is needed in order to collapse the four correlated macrovariables for netX and netY to the two known causal macrovariables for this synthetically generated dataset.

2.4 Causal Model Drift Detection and Adaptation

- **Goal:** Monitor uncertainty of data-driven causal models and concepts and differentiate between data and concept drift, thereby minimizing the need for model retraining
- **Accomplishments:** Developed Python 3 software package for CausAnom, which
  - Ensures robustness to data drift of environmental variables, thereby reducing the need for system retraining and corresponding algorithm maintenance costs.
  - Tests our methods for causal explanation of anomalous (counterfactual) data, and model uncertainty estimation on a real-world ship power prediction dataset [Andrey23].

2.4.1 Methods

In [Strelnikoff23], we introduce CausAnom, a method for causality-based anomaly detection and attribution. Specifically, given time-series data of causal variables of interest, we leverage the theory of independent causal mechanisms to estimate a local functional relationship between each pair of causally related variables in the estimated "causal graph". Given this (now) quantitative causal graphical representation, our goal is to estimate the probability of counterfactual scenarios by fitting the learned model to the data. This enables us to determine the "best-fit" explanatory model for any observed deviations, e.g., counterfactuals, anomalies, or dataset drift.

Following equation 5 in [Strelnikoff23], we restrict the functional relationships in the graph such that each "effect" (or "child") node is estimated as an additive linear

combination of "cause" (or "parent") nodes. This additive relationship enables us to remove the effects of specific parent nodes in the graph on their children in an independent manner. These removals simulate "interventions" on the graph [Pearl09] which produce counterfactual scenarios. These counterfactual scenarios, described by quantitative causal graphs with specific missing edges, can be fit to the observed data in cases of anomalies or dataset drift to explain why the data is anomalous. In this way, the CausAnom framework can not only detect anomalies (as deviations from the expected causal behavior) but also enables us to perform causality-based anomaly attribution for a class of anomalies resulting from broken cause-effect relationships. In Section 2.4.2.2, we describe the results of this procedure on the "Shifts Marine Cargo Vessel Power Consumption Prediction Dataset" [Andrey23].

2.4.2 Evaluation

2.4.2.1 Anomaly Detection Robustness

First, we test CausAnom using a pendulum system. Specifically, we consider a causal system consisting of a pendulum, sun, and ground plane in which the features of interest are the sun angle, pendulum angle, shadow position, and length. The pendulum simulation and corresponding causal graph are shown in Figure 17.

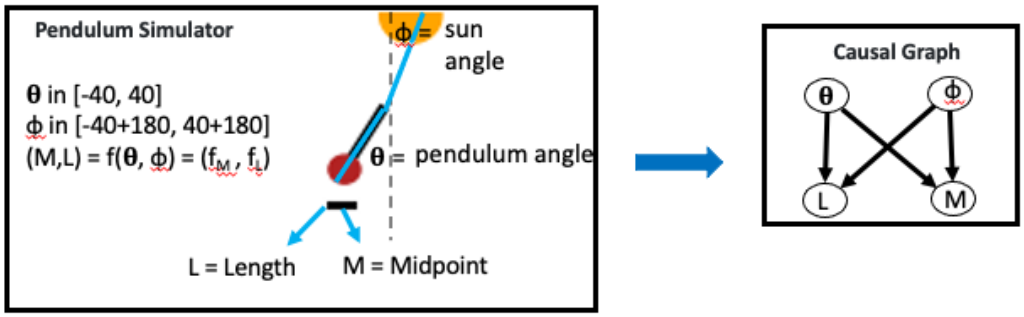

Figure 17. Pendulum simulation and corresponding causal graph

As described above, CausAnom leverages the causal graph by modeling the effect variables as a function of their causes. This differs from statistical joint anomaly detection schemes, which model the joint distribution between all variables without consideration for the underlying causal relationships. One benefit of the CausAnom approach is that it is more robust to training dataset sample bias, since it is required to only model the conditional distribution between each variable and its causal parents, rather than the full joint distribution. For example, if the pendulum training data is biased to contain only examples in which the pendulum is swinging near equilibrium (hanging vertically) any system data where pendulum deviates more significantly from equilibrium would be identified as anomalous by a non-causal model. Note however that the causal relationships as depicted in Figure 17 remain unviolated for the testing data, so CausAnom would not erroneously identify these unseen states as anomalous.

We validate this claim with the following experiment:

Case 1 defines a standard scenario in which the training data is not biased. The training data and nominal testing data each consist of non-overlapping random samples from the nominal system for which the pendulum angle is within ±40° and the sun angle is

within (±40+180)°. To generate synthetic anomalies, we artificially modify the effect of the pendulum on the shadow variables by shifting the pendulum by an angle θ_0. We consider values of θ_0 between 10° (mild anomaly) and 80° (strong anomaly) to test the efficacy of CausAnom in diverse regimes.

In case 2, we instead train CausAnom on biased data, where the pendulum angle is only between -40° and 40°, however we test on nominal data containing pendulum angles between 40° and 50°. For both cases, in addition to the CausAnom anomaly detection scores, we report scores for a baseline autoencoder which models the full joint distribution without consideration for the causal structure. Because these methods all require a threshold to be set to define an anomalous state, we report the F1 scores corresponding to the highest scoring threshold. Figure 2 shows the results of this experiment.

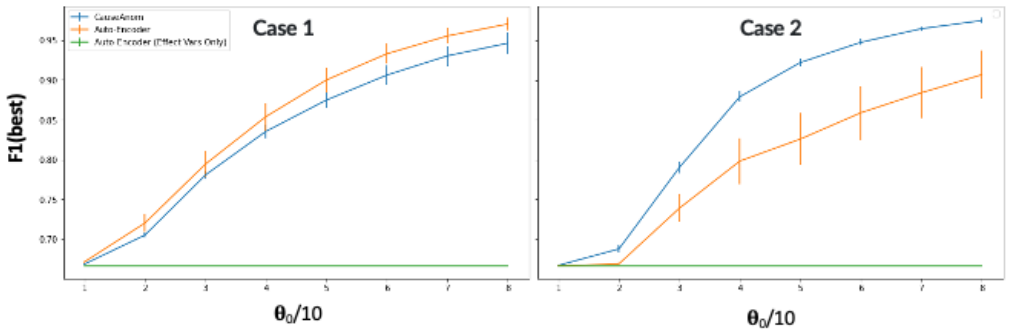

Figure 18. Best F1 scores for CausAnom, and the baselines: auto-encoder, and auto-encoder (with effect vars only). Case 1 models the standard training scenario and Case 2 models the biased training data scenario.

From this we note that, in Case 1 (standard training scenario) CausAnom performs similarly to the auto-encoder baseline, which is an expected result since this scenario does not benefit from the modeling of the causal graph. On the other hand, Case 2 (biased training scenario) shows a clear dominance by CausAnom over the baseline method via the reduction of false positive anomalies.

2.4.2.2 Anomaly Attribution and Uncertainty Estimation

For the following experiments, we leverage Shifts Marine Cargo Vessel Power Consumption Prediction Dataset [Andrey23] ("Ship Power dataset", a realistic dataset consisting of measurements sampled every minute from sensors on-board a merchant ship over a span of 4 years. For the following, we analyze the dynamics and causal factors associated with the power of the ship.

To validate the broken-link anomaly attribution scheme in this realistic scenario, we assume early data is "nominal" and later data is "off-nominal" since it is anticipated that deviations appear over time due to hull fouling and sensor drift. To estimate the causal deviation between the nominal/off-nominal regimes, we estimate the causal graph using the data in each of the respective regimes. The results of that are shown in Figure 4.

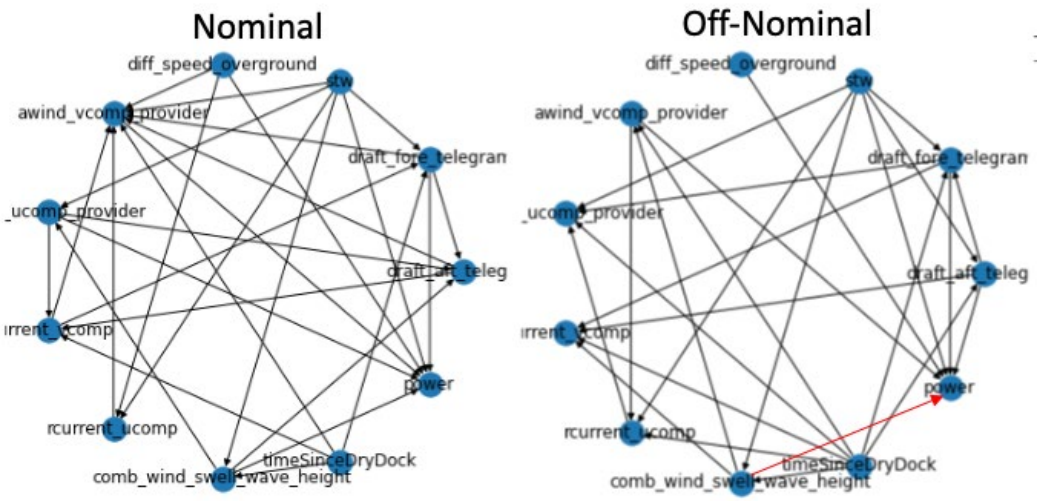

Figure 19. Nominal (left) and Off-nominal (right) causal graphs for the Ship Power dataset.

Specifically, we note that the power variable has a missing link from the swell wave height variable in the off-nominal graph. By training CausAnom with the nominal data and causal graph, we were able to correctly identify this broken causal link, providing an explanation for the model's anomalous behavior and potential diagnostic hints.

The dataset analyzed above is meant to test the native uncertainty estimation capabilities of ship power prediction models. Specifically, given the learned CausAnom model on this dataset, we aim to estimate how closely the CausAnom uncertainty estimates match the actual model errors. This is done by ordering the predictions in ascending order according to their predictive error and calculating the area under the cumulative mean-squared error (MSE) curve (smaller area is better). Figure 4 shows the plots for this metric where we have also compared against an ensemble multilayer perceptron (MLP) method. Note, CausAnom performs significantly worse, with the area under the curve (R-AUC) of R-AUC = 1.0 X 10^6. The baseline model achieves R-AUC = 4.5 X 10^5. This baseline, however, does not utilize the causal graph or any additional feature selection, so it can use predictive information from all variables, whereas CausAnom only leverages the causal parents associated with ship power. We hypothesize that this restriction is the primary factor in the poor performance of CausAnom, and indeed we observe that the baseline model's performance reduces to 1.6 X 10^6 when restricting its dependent variables to only the causal parents. We offer two possible explanations for these results. First, the causal graph may be incorrectly estimated (and the estimated collection of causal parents of the power variable may be incomplete) implying that important causal factors of power are ignored by the CausAnom model. The second possibility is that the causal graph is in fact complete, however the non-causal (causal) variables have simple (complex) functional relationships to the power. This would mean that, while CausAnom could theoretically match the accuracy of the baseline model, it is much more difficult to learn a model based on the true causal parents than it is to simply leverage the information from the non-causal variables.

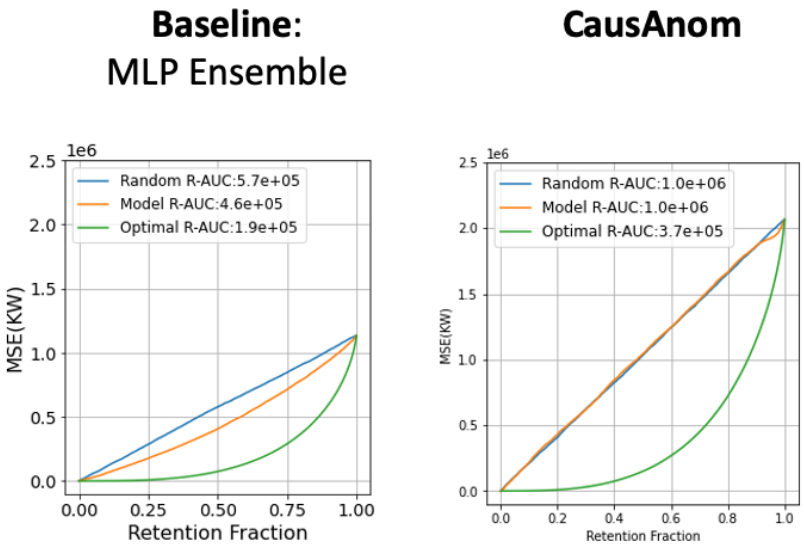

Figure 20. Area under MSE (R-AUC) for CausAnom and MLP ensemble baseline. This metric estimates how well the model uncertainty matches the prediction accuracy in experiments on the Ship Power dataset.

Recommendations for Future Technology and Research

We detailed our approach for the various software modules that comprise the CADA system. It should be noted that the existence of cause-and-effect concepts in any system of high-dimensional variables is not guaranteed. However, we have shown that in datasets that do contain cause-and-effect concepts, it is possible to utilize some amount of human supervision to enhance the correlation-based representations of discovered concepts such that they more concisely represent the known causal dynamics of the system.

Among the lessons learned on this project, we have found that one of most challenging aspects of causal inference and modeling research is finding real-world observational datasets with known causal ground truth. This is a major limitation in performing the type of thorough testing and evaluation that would enable transitioning cutting-edge causal theory to applications of naval relevance. Despite this issue, we are optimistic about the future of applied causal inference due recent calls for benchmark datasets (e.g., https://www.cclear.cc/2023/CallforDatasets) and real-world applications (e.g., https://causal-machine-learning.github.io/kdd2023-workshop/cfp/). Along these lines, we partition our recommendations into those for smoother transition of ONR-funded research projects into future naval capabilities, and interesting technical next steps in the research areas we explored during this program.

Recommendations for smoother transition:

- Glenn White's guidance in the proposal and early stages of this project was indispensable for us. If possible, assign a naval subject matter expert as a point-of-contact to each performer in case they have questions about naval relevance or desire quick subjective evaluations of their work.
- If possible, make unclassified simulation platforms (e.g., versions of AFSIM, JSAF) and navy-relevant all-source/multi-INT datasets available to performers to avoid "reinventing the wheel" and provide a clear benchmark.

Technical Next Steps:

- **Symbolic Temporal Reasoning.** As the simulation environment increases in fidelity, so too will the number of symbols generated at run time and the number of rules that process the observed symbols. This offers us the opportunity to tune the amount of uncertain information and determine the effects on causal learning.
- **Causal Feature Learning.** Human subject research verifying lower cognitive load due to analyzing concepts as opposed to high-dimensional data is needed. We would also like to explore the tradeoff in accuracy/number/complexity of concepts with respect to the amount of supervision needed to obtain them.
- **Drift Detection and Adaptation.** We are currently exploring improvements to the uncertainty estimation abilities of CausAnom. Specifically, we seek to model chained uncertainty estimates and their propagation through the causal graph to improve upon the results presented in Section 2.4.2.2.

Breakdown of all Contract Costs

Contract No	N00014-19-C-2024/004				Preparation Date:		6/23/2023			
Contract Type:					Reporting Period From:		4/29/2023	Thru:	5/26/2023	
18016			Funds & Labor Hour		Period of Performance:		7/29/2022	Thru:	7/28/2023	
BH4E			Expenditure		Funding:	$ 498,042				
	CADA Option 3		Report		HRL PROPRIETARY					
	A		B		C		D		E	
	ORIGINAL NEGOTIATED CONTRACT		LATEST NEGOTIATED CONTRACT CHANGES		REPORTING PERIOD EXPENDITURES		CUMULATIVE EXPENDITURES		ESTIMATE TO COMPLETION	
	A1	A2	B1	B2	C1	C2	D1	D2	E1	E2
	MAN HOURS	DOLLAR AMOUNT	MAN HOURS	DOLLAR AMOUNT	MAN HOURS	DOLLAR VALUE	MAN HOURS	DOLLAR VALUE	MAN HOURS	DOLLAR VALUE
CA	60	3,780	60	3,780	0	0	0	0	60	3,780
CA1	0	0	0	0	0	0	1	23	(1)	(23)
MTS	780	66,105	780	66,105	0	0	0	0	780	66,105
PA1	0	0	0	0	3	129	17	673	(17)	(673)
SMTS1	380	41,682	380	41,682	0	0	0	0	380	41,682
T4	0	0	0	0	28	2,421	700	59,241	(700)	(59,241)
T5	0	0	0	0	28	3,107	777	87,145	(777)	(87,145)
TE	740	32,140	740	32,140	0	0	0	0	740	32,140
Total Hours & Dollars	1,960	143,707	1,960	143,707	59	5,657	1,494	147,082	466	3,375
OVERHEAD		262,551		262,551		9,319		260,269		2,282
TOTAL LABOR & O/H		406,259		406,259		14,975		407,351		(1,092)
MATS & PARTS		0		0		0		0		0
TRAVEL EXPENSES		0		0		0		0		0
ODC,SUB		2,677		2,677		0		973		1,704
SUB-TOTAL COSTS		408,936		408,936		14,975		408,324		612
G&A		38,005		38,005		623		19,781		18,224
COST OF MONEY		10,876		10,876		2,083		4,897		5,979
TOTAL COST		457,817		457,817		17,681		433,002		24,815
FEE / PROFIT @9.00		40,225		40,225		1,404		38,529		1,695
TOTAL CPFF		498,042		498,042		19,084		471,532		26,510
P.O. Commitments								0		
TOTAL COMM & EXP						19,084		471,532		

Distribution: Authorized to US Government agencies only.

Distribution authorized to U.S. Government Agencies and private individuals or enterprises eligible to obtain export-controlled technical data in accordance with DoD 5230.25.

References

[Agmon11] Noa Agmon, Daniel Urieli, and Peter Stone. Multiagent Patrol Generalized to Complex Environmental Conditions. In Proceedings of the Twenty-Fifth Conference on Artificial Intelligence (AAAI), August 2011.

[Andrey23] M. Andrey, et al. "Shifts Marine Cargo Vessel Power Consumption Prediction Dataset," site: https://zenodo.org/record/7684813#.ZFvghezMLm0

[Besold17] Besold, T. R., Garcez, A. D. A., Stenning, K., van der Torre, L., & van Lambalgen, M. (2017). Reasoning in non-probabilistic uncertainty: Logic programming and neural-symbolic computing as examples. Minds and Machines, 27(1), 37-77.

[Chalupka16] Chalupka K, Bischoff T, Perona P, Eberhardt F (2016) Unsupervised discovery of el Niño using causal feature learning on microlevel climate data. In: Proceedings of the thirty-second conference on uncertainty in artificial intelligence.

[Chalupka17] Chalupka, Krzysztof et al. "Causal feature learning: an overview." Behaviormetrika 44.1 (2017): 137-164.

[Guyon19] Isabelle Guyon, Alexander Statnikov, and Berna Bakir Batu. Cause Effect Pairs in Machine Learning. Springer, 2019.

[He15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In CVPR, 2015.

[He17] Kaiming He, Georgia Gkioxari, Piotr Dolla´r, and Ross Girshick. Mask R-CNN. In ICCV, 2017.

[Jammalamadaka23] A. Jammalamadaka, L. Zhang, J. F. Comer, T.-C. Lu and R. Bhattacharyya, "Semi-Supervised Learning of Visual Causal Macrovariables," in The 36th International FLAIRS Conference (FLAIRS-36), 2023. (*accepted*)

[Johnson17] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In CVPR, 2017a.

[Landy14] Landy, D., Allen, C., & Zednik, C. (2014). A perceptual account of symbolic reasoning. Frontiers in psychology, 5, 275.

[Lundberg20] Lundberg, S.M., Erion, G., Chen, H. *et al.* From local explanations to global understanding with explainable AI for trees. *Nat Mach Intell* **2,** 56–67 (2020). https://doi.org/10.1038/s42256-019-0138-9.

[Manhaeve19] Manhaeve, R., Dumancic, S., Kimmig, A., Demeester, T., & De Raedt, L. (2018). Deepproblog: Neural probabilistic logic programming. Advances in Neural Information Processing Systems, 31, 3749-3759.

[Mao19] Mao, J., Gan, C., Kohli, P., Tenenbaum, J. B., & Wu, J. (2019). The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision. arXiv preprint arXiv:1904.12584.

[Marx18] Marx, A & Vreeken, J Causal Inference on Multivariate and Mixed Type Data. In: Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Data (ECMLPKDD), Springer, 2018.

[Pearl09] Pearl, Judea. Causality. Cambridge university press, 2009.

[Selvaraju17] Selvaraju, Ramprasaath R., et al. "Grad-cam: Visual explanations from deep networks via gradient-based localization." *Proceedings of the IEEE international conference on computer vision*. 2017.

[Strelnikoff23] S. Strelnikoff, A. Jammalamadaka, and T.-C. Lu, "Causanom: Anomaly DetectionWith Flexible Causal Graphs," in The 36th International FLAIRS Conference (FLAIRS-36), 2023. (*accepted)*

[Uziel20] Uziel, Steven J. AI-Augmented Decision Support Systems: Application in Maritime Decision Making Under Conditions of METOC Uncertainty. Diss. Monterey, CA; Naval Postgraduate School, 2020.

[Vila94] Vila, L. (1994). A survey on temporal reasoning in artificial intelligence. Ai Communications, 7(1), 4-28.

Appendices

6.1 CADASIM "Red Rover" Scenario and example code

Here we describe a simple CADASIM scenario which we have used to prove our causal feature learning methods. The scenario consists of three entities: BLUE (ego ship), RED (adversarial ship) and GRAY (neutral ship). RED missions are to patrol specific areas (defined by a radius as shown in Figure 21) and chase if BLUE crosses perimeter. BLUE mission is to reach destination on right hand side. BLUE COAs ("aggressive" and "cautious") are parameterized by the {distance threshold, closing angle} for considering RED a threat. This is an example of how risk averse decision-makers can be modeled. Utility depends on time to complete mission and time spent within firing distance of RED. GRAY does not interfere with RED or BLUE, however GRAY's location correlates with the location of RED during the historical observation period. This correlation could hypothetically be due to other potential correlates like land masses, weather patterns, or additional agents with uncertain flags or locations.

The code snippet in Figure 22 was used to implement all but the gray ship in this scenario, showcasing the (relative) simplicity of creating and running simulations in CADASIM. It illustrates the creation of navigation functions by calling higher-order

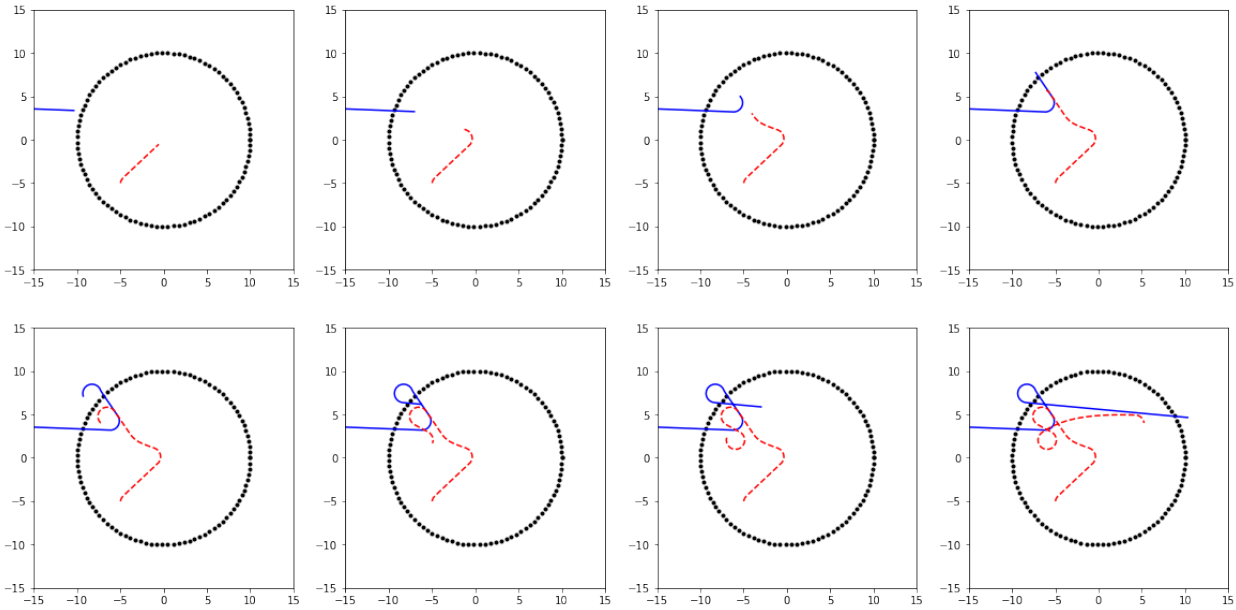

Figure 21. Example scenario involving RED and BLUE ships/agents. Starting from the top-left frame and moving toward the bottom-right, the BLUE ship is tasked with moving to a position far in the east. As soon as BLUE enters the security perimeter guarded by the RED ship, RED begins to intercept BLUE. Once the BLUE ship determines that they are being intercepted by RED, they take evasive maneuvers, ultimately leading them to backtrack out of the security perimeter. Once BLUE is outside the security perimeter, RED begins returning to the center of their area. But, realizing that they are no longer being threatened, BLUE resumes their course eastward, again crossing into the security area. Immediately, RED returns to their pursuit of BLUE, but this time, they are behind BLUE, enabling BLUE to travel eastward through (and then out of) the security area.

function generators with some parameters (e.g., `nav_guard_fn` takes x and y locations for the center of the security area, the two radius parameters r_0 and r_1, and the designation of "friendly" agents that will not be intercepted) and creating some ship objects (e.g., a `WarShip` class is initialized with x and y coordinates, a heading, a maximum turning rate, a maximum speed, a Flag designation, a `ShipClass`, a radar cross-section value, and a navigation function). The `SimpleWorld` object functions as a container and driver for the simulation. When the simulation runs, the state variables for every entity are recorded into Python 3 pandas data frames, allowing them to be easily serialized to disk or passed to another application.

```
# Initialize Red
red_nav_fn = nav_guard_fn(0., 0., 0.01, 1, 10, Flag.RED)
red = WarShip(-5., -5., np.pi/2, np.pi/10, 20./60, Flag.RED,
              ShipClass.DESTROYER, 2, red_nav_fn)
red_radar = SimpleRadar(_, _, 5., 0.5, platform=red)
red.add_mounted_sensors([red_radar])

# Initialize Blue
blue_nav_fn = nav_direct_timid_fn(50., 1., 0.01, 1., 4., np.pi/4,
Flag.BLUE)
blue = WarShip(-50., 5., 0., np.pi/10, 20./60, Flag.BLUE,
               ShipClass.DESTROYER, 2, blue_nav_fn)
blue_radar = SimpleRadar(_, _, 5., 0.5, platform=blue)
blue.add_mounted_sensors([blue_radar])

# Create and run simulation
world = SimpleWorld(red, blue, red_radar, blue_radar)
T = 8*60 # 8 hours
world.run(T)
```

Figure 22. Code snippet showcasing ease of naval scenario creation in CADASIM.

6.2 TMPLAR Human Subject Experiment Data

Feature	Description
Age	Age of the subject
Air_Temp_Max	The maximum air temperature evaluated against all other weather forecasts using the maximum air temperature among all way points in a given route
Air_Temp_Median	The median air temperature evaluated against all other weather forecasts using the maximum air temperature among all way points in a given route
Air_Temp_Percentile	90th percentile value of air temperature evaluated against all other weather forecasts using the air temperature among all way points in a given route
Bearing Normal	normalized sum over the changes in bearing (from previous bearing) for the entire route
Bearings	Sequence of bearing for the chosen route
Confidence	Confidence in the chosen route on a 1 – 7 Likert Scale
Dist Chosen	Total distance for the chosen route

DSS	Decision Support System level
Fuel Chosen	Fuel consumption for the chosen route
Fuel max	Maximum fuel possible given all the weather scenarios for the chosen route
Fuel median	Median fuel possible given all the weather scenarios for the chosen route
Fuel percentile	90th percentile value of the fuel cost that is within weather limit
Gender	Gender of the subject
Ht_Sig_Max	The maximum significant wave height evaluated against all other weather forecasts using the maximum significant wave height among all way points in a given route
Ht_Sig_Median	The median significant wave height evaluated against all other weather forecasts using the maximum significant wave height among all way points in a given route
Ht_Sig_Percentile	90th percentile value of significant wave height evaluated against all other weather forecasts using the maximum significant wave height among all way points in a given route
Ht_Swell median	The median swell height evaluated against all other weather forecasts using the maximum swell height among all way points in a given route
Ht_Swell_Max	The maximum swell height evaluated against all other weather forecasts using the maximum swell height among all way points in a given route
Ht_Swell_percentile	90th percentile value of swell height evaluated against all other weather forecasts using the maximum swell height among all way points in a given route
NumRoute	Number of Routes on the Pareto front
Points	Waypoints of the chosen route
Q1 Click	Number of click in quadrant 1
Q1 View	Total percentage time spent in quadrant 1
Q2 Click	Number of click in quadrant 2
Q2 View	Total percentage time spent in quadrant 2
Q3 Click	Number of click in quadrant 3
Q3 View	Total percentage time spent in quadrant 3
Q4 Click	Number of click in quadrant 4
Q4 View	Total percentage time spent in quadrant 4
Route ID	Route ID for a given scenario
Route Name	Starting location to End location
Route Time	Time consumption for the chosen route
RWspd_Max	The maximum relative wind speed evaluated against all other weather forecasts using the maximum relative wind speed among all way points in a given route
RWspd_Median	The median relative wind speed evaluated against all other weather forecasts using the maximum relative wind speed among all way points in a given route
RWspd_Percentile	90th percentile value of relative wind speed evaluated against all other weather forecasts using

	the relative wind speed among all way points in a given route
Sample #	Integer
Scenario	Scenario ID, total 96 scenarios
Score	the normalized Manhattan distance between the objective values for a given route and the objective values of a "utopian" route, which is generated by choosing the best distance and the best fuel cost from all the routes in a given scenario
Time	Total time for the test (there are 0 in this column)
User ID	
Wspd_Max	The maximum wind speed evaluated against all other weather forecasts using the maximum wind speed among all way points in a given route
Wspd_Median	The median wind speed evaluated against all other weather forecasts using the maximum wind speed among all way points in a given route