

DARPA GAILA Phase III: Milestone 6 Report

Advanced Cognitive Learning for Embodied Language - AILEEN

Prepared by Shiwali Mohan
with inputs from Yonghui Fan and William Hancock

September 27th 2022

1 Summary

In our previous milestone report, we described preliminary results for learning object classes and attributes as well as for events. In this report, we conclude our research efforts on GAILA with updates

1. Updated visual learning and reasoning pipeline: We present results from an updated design of the visual reasoning system focusing on 1) continual learning and 2) automatic discovery of geometric properties of the objects from images.
2. Updated action and event representations: We present results for learning inspectable, event representations from demonstrations.

We presented a demo of our system to the PM team in the PI meeting in August 20223

2 New Continual Learning and Concept Discovery Pipeline

In this phase, we focus on learning new objects and discovering their distinguishing concepts AILEEN's world. We report results from two explorations:

1. Can we learn new objects when it appears for the first time? Specifically, can we learn a new object category with only one sample?
2. Can we discover distinguishing geometric properties of objects from the newly learned object?

2.1 New Continual Learning Pipeline in Visual Module

Our research along this thrust has one primary goal: we want to investigate if AILEEN can learn new objects autonomously, with only few observed samples. Towards this target, we stress the following requirements: (1) how to automatically add a new object to the shape knowledge without updating the original feature extractor? (2) how to continuously learn new objects without catastrophically forgetting the previously learned knowledge? (3) since all AILEEN shapes are placed in the world with random locations and orientations, the samples from the same category may have different 2D looks in the scene. How to deal with the different varieties of the new category with one observed sample? (4) Besides the possible exterior differences of the same object in a 2D image, is the pipeline sensitive to different internal textures of the object?

The pipeline of AILEEN is shown in Figure 1. We start by training a VGG16 network for two known objects "box" and "sphere" as our base knowledge of shapes. This trained VGG16 network also serves as the feature extractor and remains unchanged throughout all the learning tasks. When a new object, for instance, a "cone", appears, AILEEN will give a wrong prediction for sure based on the current base knowledge. The

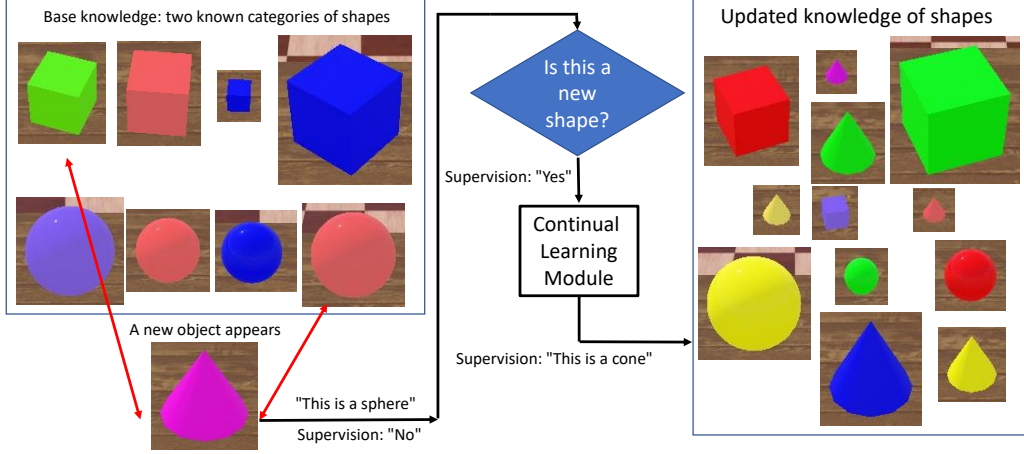


Figure 1: Pipeline of AILEEN Continual Learning Module.

supervision gives a correction and instructs the student that this is a new shape, which triggers the continual learning module to add this new shape to the knowledge of shapes and finish learning a new shape category.

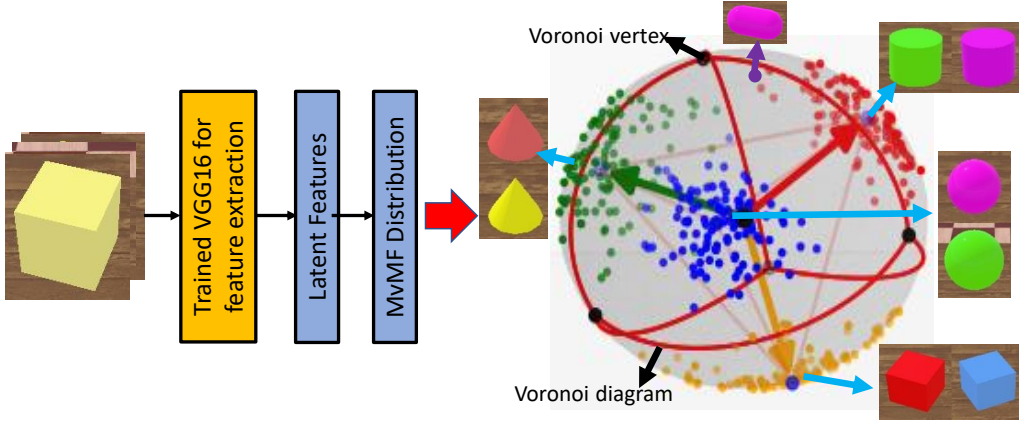


Figure 2: Pipeline of Updating MvMF Model.

The core of AILEEN continual learning module is a Mixture of Von-Mises Fisher distribution (MvMF). In the beginning, we use features of “box” and “sphere” objects to initialize an MvMF and get two cluster centers for each known shape in a unit sphere space. When the continual learning module is activated, a hundred pseudo data points will be uniformly sampled around each existing cluster. We duplicate the feature of the new object by one hundred times and concatenate them with the features of pseudo data points to obtain a mini dataset. The MvMF distribution is updated by using this mini dataset, and a new cluster center will be generated for the new class. The process of updating the MvMF model is shown in Figure 2. Each category has a unique cluster center in a unit sphere space. Figure 3 shows an example of continuously learning “cone” and “capsule” objects. By updating the MvMF model and generating new cluster centers for new categories, we realize the target of learning new objects with one shot.

The accuracy of newly learned objects is high until the “cylinder” object appears. Testing on 1000 “cylinder” cases only shows a 45.9% accuracy. We think this is caused by the gap between a large variety of “cylinder” objects and a limited available training sample. The “cylinder” object has various orientations in 2D view as shown in Figure 4(a). With one observed sample, AILEEN only recognizes “cylinder” with a specific orientation. For increasing the generalization of inferring from one sample, we use a rotation augmentation strategy.

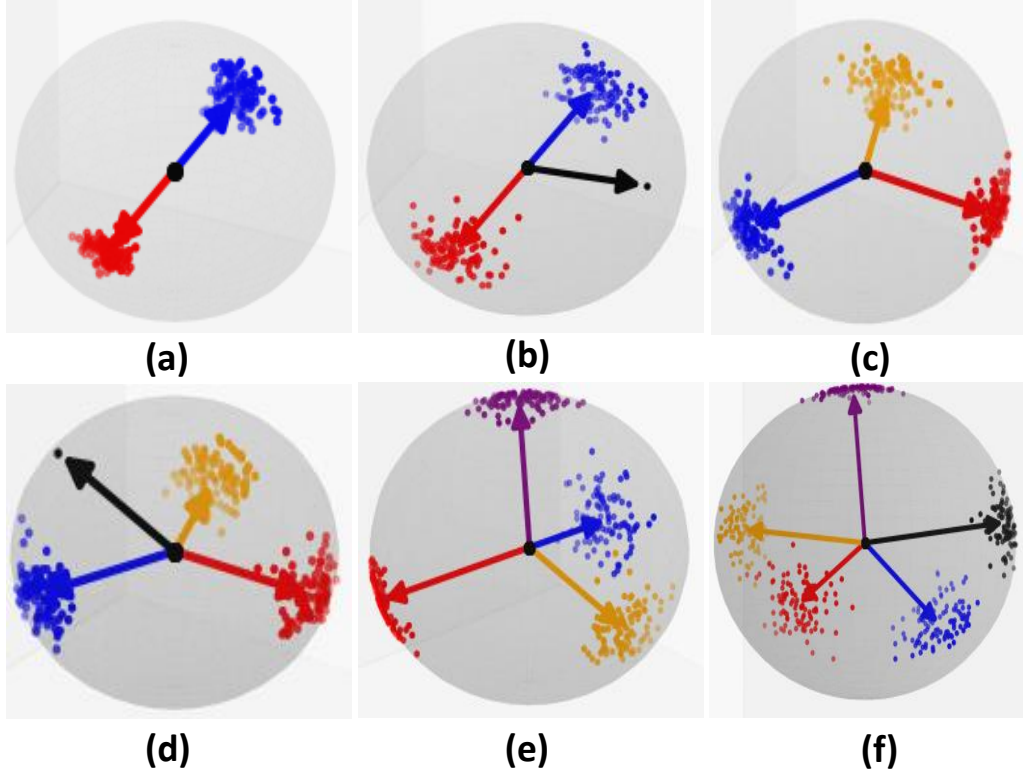


Figure 3: An example of learning “cone” and “cylinder” objects. (a) Initially, there are two cluster centers representing “box” and “sphere” in the sphere space; (b) A “cone” object appears for the first time as the black dot shows (we map the feature of “cone” to the same sphere space). The cosine distances from this “cone” sample to “box” and “sphere” cluster centers are 0.608 0.583, respectively. The AILEEN will consider this as a sphere due to a closer distance; (c) After updating the MvMF model according to Figure 2, a new yellow-colored cluster appears in the sphere space. The accuracy on 1000 testing cases is 100%. (d) A “cylinder” object appears for the first time. The distances to “box”, “sphere”, and “cone” are 0.761, 0.662, and 0.729, respectively. AILEEN will consider this as a sphere. (e) After updating the MvMF model, a new purple-colored cluster is added to the sphere space. Accuracy on 1000 testing cases is only 45.9% without augmentation. After using a simple rotation augmentation strategy, the accuracy increases to 75.8%. (f) Adding the new black-colored cluster for “soccerball”. Accuracy on 500 testing cases is 100%.

We rotate the object image every 10 degrees from 0-360° and use all these augmented samples in updating the MvMF model. This treatment improves the accuracy from 45.9% to 75.8%. This phenomenon does not happen in learning the “cone” because “cone” has a quite single variety in the AILEEN world as shown in Figure 4(b).

The above experiment shows that our current continual learning model is sensitive to the outline shape of the object. We further dig in to explore its sensitivity to internal texture changes. We test on the “soccerball” object as shown in Figure 4(c). The test on 1000 “soccerball” samples shows a 100% accuracy after learning one sample. It shows that internal texture changes may not have a significant influence on learning performance.

All the above learning tasks are designed in a continuous manner, which means we only learn one object at a time and learn the next one after updating the MvMF model for the last one. In most one-shot/few-shot learning settings, the system is required to learn multiple new objects at the same time. We repeat the learning of “cone” and “cylinder” by changing the one-by-one learning style to the learn-together style. The prediction accuracy remains the same, which means the system will reach a similar converged state no matter if the training is applied in sequence or at the same time.

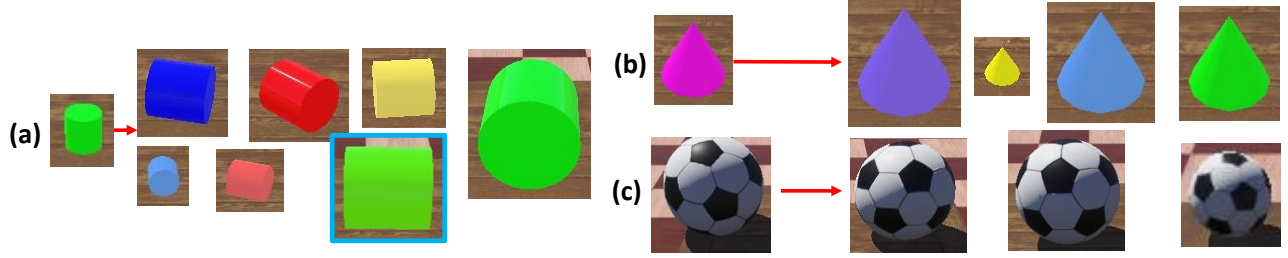


Figure 4: Shapes may have different varieties in 2D view. (a) The “capsule” has various orientations in the AILEEN world. Some of them are confusing. One observed sample is not enough to cover so many varieties. (b) The “cone” has a simple variety in 2D view, hence, the accuracy is high even though only one sample is used to train. (c) The contour of “soccerball” is identical but each “soccerball” has different internal patterns in 2D view.

We also test on different feature extractors. We take VGG16 as the backbone and test on different initialized numbers of known categories. Still, these changes have no obvious influence on the convergence of the final MvMF model.

In summary, our continual learning pipeline is able to sequentially or parallelly learn new objects. For objects with changeable looks in 2D view, a data augmentation strategy will greatly help to increase the generality of our method. A feature-level augmentation may be a better solution to this problem.

3 Discovery of Geometric Features

In our previous report, we demonstrated that using cosine distance as a metric is able to measure the representativeness of a geometric feature to each category. In this series of experiments, we apply the same method to both old and new objects to verify the effectiveness of discovering meaningful concepts from each category. The primary difference between this work and our previous experiment is that the cluster centers of “cone” and “cylinder” are learned with only one sample. As a comparison, we will show the results from a 1-sample learned model and the results of a model trained in a supervised learning manner with 500 samples.

We first test discovery relevant geometric features automatically. We use two methods to extract geometric features from one shape per category: first, 40 concepts are extracted by using conventional patching methods; second, a number of geometric features are extracted by using super-pixel patching methods. We compared SLIC, WaterShed, QuickShift, and Felzenszwalb’s methods and chose SLIC as our super-pixel patching method. The total numbers of geometric features for each category are: “box”: 181, “sphere”: 168, “cone”: 72, “capsule”: 184.

Figure 5 shows the top five representative concepts of “box” and “sphere” shapes selected by the cosine distance between the concept to the cluster center. The closest distance is marked by blue and the second closest distance is marked by green. In these two examples, 1-sample results are consistent with 500-samples results. Figure 6 shows the top five representative concepts of “cylinder” and “cone” shapes. Noticing that the selections of 1-sample and 500-sample are not completely identical. But, empirically, all these concepts are unique in “cylinder” objects. The results on “cone” show some limitations of our current solution. These concepts are expected to have the closest distance to the “cone” cluster center. However, all these selected concepts actually have the closest distance to either “box” or “cylinder” cluster centers. And in this example, the distances from 500-sample results seem more reasonable.

We then test some specific concepts we care about: “line”, “flat plane”, “corner” and “curve”. Figure 7 shows the distances of “line” and “flat plane” concepts to each cluster center. Noting that, although “capsule” has no line, the 2D view of a “capsule” may have line parts in the contour. Similarly, the “cone” also has line parts in its 2D view. In the results of “line” concepts, it is correct that “box” and “cylinder” have a closer distance. The





	1 sample					500 samples per shape				
Geometric Structures										
Sphere	1.024	0.990	0.952	1.009	0.982	0.872	0.866	0.830	0.869	0.924
Cylinder	0.822	0.815	0.799	0.808	0.896	0.765	0.776	0.813	0.841	0.860
Box	0.640	0.665	0.701	0.733	0.795	0.653	0.691	0.712	0.721	0.855
Cone	0.929	0.860	0.883	0.793	1.030	0.834	0.807	0.806	0.777	0.970
										
Sphere	0.733	0.809	0.836	0.868	0.876	0.732	0.744	0.813	0.816	0.827
Cylinder	0.817	0.766	0.801	0.830	0.845	0.821	0.729	0.777	0.745	0.746
Box	0.997	0.890	0.945	0.878	0.878	0.906	0.820	0.878	0.833	0.841
Cone	0.977	0.968	0.963	1.009	1.008	0.885	0.896	0.925	0.941	0.943

Figure 5: Top five concepts of “box” and “sphere”. These concepts are considered the most representative component of the shape. The results of using 1 sample and 500 samples are given to show consistency.

“cone” is supposed to have a closer distance than the “sphere”, but we see the opposite results. The “flat plane” results are generally reasonable. Figure 8 shows the distances of “corner” and “curve” concepts to each cluster center. Being similar to the “line” concept, “capsule” and “cone” objects also have “corner” concepts in their 2D view. The results on “corner” are correct on “box” and “capsule”, but unsatisfied on “cone” and “sphere”. The results of the “curve” concept on the 500-sample model are partially correct. However, the results of the 1-sample model are considered incorrect.

We also test on different feature extractors. Similarly, we take VGG16 as the backbone and test on different initialized numbers of known categories. Still, these changes have no obvious influence on the concept discovery experiments.

In summary, our current method can automatically answer the question “what are the unique parts in the object?”, but it shows weakness in answering the questions like “does this object has a curve?”. Actually, we never teach what a “curve” is to AILEEN. A link between discovered components and their attributes is necessary for future work. Also, we can see that the patch boundary can cause influences in the metric space. A similar concept with different cutting boundaries may have totally different distances to cluster centers.

4 Action and Event Representations

Recall that in our M5 report, we discussed updates to the action/event representation scheme such that AILEEN can learn concepts that correspond to the *manner* of an action in addition to the *pre/post* conditions of actions. In this section, we describe implementation of a new representational scheme and present preliminary results from our explorations. Specifically, we build upon qualitative representation research to develop bounded spatio-temporal descriptions of a scene. These representations describe how attributes of and relations between objects change over time. Our results are promising and demonstrate that our cognitive robotic system AILEEN can learn a richer conceptual definition of events from few examples.

	1 sample					500 samples per shape				
Geometric Structures										
Sphere	0.931	0.950	0.986	0.988	0.961	0.866	0.907	0.887	0.873	0.915
Cylinder	<u>0.772</u>	<u>0.806</u>	<u>0.825</u>	<u>0.850</u>	<u>0.856</u>	<u>0.588</u>	<u>0.758</u>	<u>0.773</u>	<u>0.819</u>	<u>0.827</u>
Box	<u>0.794</u>	<u>0.818</u>	<u>0.713</u>	<u>0.778</u>	<u>0.774</u>	<u>0.829</u>	<u>0.770</u>	<u>0.830</u>	<u>0.666</u>	<u>0.842</u>
Cone	0.967	0.995	0.917	1.005	0.954	0.905	0.910	0.924	0.865	0.919
Sphere	0.962	1.058	1.016	0.965	0.979	0.787	0.867	0.892	0.920	0.888
Cylinder	<u>0.640</u>	0.757	<u>0.690</u>	<u>0.866</u>	<u>0.799</u>	0.740	0.800	<u>0.633</u>	<u>0.856</u>	<u>0.698</u>
Box	<u>0.608</u>	<u>0.570</u>	<u>0.680</u>	0.916	<u>0.820</u>	<u>0.613</u>	<u>0.577</u>	<u>0.724</u>	0.894	<u>0.756</u>
Cone	0.733	<u>0.739</u>	0.868	<u>0.886</u>	0.894	<u>0.608</u>	<u>0.647</u>	0.828	<u>0.883</u>	0.893

Figure 6: Top five concepts of “cylinder” and “cone”.

	1 sample				500 samples per shape			
Geometric Structures								
Sphere	0.998	0.984	0.979	0.901	0.953	0.943	0.942	0.822
Cylinder	<u>0.942</u>	<u>0.956</u>	<u>0.955</u>	<u>0.811</u>	<u>0.880</u>	<u>0.873</u>	<u>0.864</u>	0.840
Box	<u>0.868</u>	<u>0.883</u>	<u>0.890</u>	<u>0.826</u>	<u>0.863</u>	<u>0.869</u>	<u>0.875</u>	<u>0.776</u>
Cone	1.019	1.005	1.020	0.916	0.988	0.983	0.986	<u>0.808</u>
Sphere	1.000	0.973	1.009	0.982	0.879	0.841	0.869	0.924
Cylinder	<u>0.748</u>	<u>0.734</u>	0.808	<u>0.896</u>	<u>0.691</u>	<u>0.768</u>	0.841	<u>0.860</u>
Box	<u>0.680</u>	<u>0.687</u>	<u>0.733</u>	<u>0.795</u>	<u>0.731</u>	<u>0.729</u>	<u>0.721</u>	<u>0.854</u>
Cone	0.949	0.902	<u>0.793</u>	1.030	0.919	0.811	<u>0.777</u>	0.970

Figure 7: Cosine distances of four “line” and “flat plane” concepts to each cluster center.





	1 sample				500 samples per shape			
Geometric Structures								
Sphere	0.967	0.978	0.967	0.972	0.921	0.944	0.925	0.918
Cylinder	0.914	0.953	0.918	0.926	0.858	0.885	0.875	0.855
Box	0.861	0.896	0.868	0.858	0.841	0.873	0.853	0.883
Cone	1.006	1.021	0.967	0.997	0.979	0.987	0.961	0.977
								
Sphere	0.876	0.935	0.935	0.875	0.688	0.902	0.788	0.816
Cylinder	0.871	0.877	0.883	0.833	0.770	0.890	0.817	0.778
Box	0.822	0.852	0.867	0.780	0.777	0.865	0.804	0.759
Cone	0.936	0.978	0.982	0.962	0.851	0.964	0.882	0.953

Figure 8: Cosine distances of four “corner” and “curve” concepts to each cluster center.

4.1 Updated Event Representations

In the previous phase, we showed that AILEEN could learn how to place an object relative to another, e.g. ‘place the red sphere north of the green cylinder’. We are now working towards demonstrating that AILEEN can learn a more diverse array of event and action concepts. Currently we are focusing on learning ‘push’, ‘pull’, ‘drop’, and ‘pick-up’. To do this, we are introducing temporally bounded representations that describe how quantities are changing over specified intervals. These intervals correspond to changes in the world, e.g. the period in which two objects are getting closer, or the period in which an object is falling. We relate these quantity changes using Allen’s Interval Algebra (AIA) to construct an event case that can be learned analogically.

4.2 Quantity Encoding

We define a set of quantities and corresponding encoding methods that define what information is included in cases. We utilize QSRLib’s [2] rcc8, cardir, and moving or stationary qualitative spatial relations. Additionally, this work introduces two new encodings, the *sign of the derivative* and the. This encoding maps a continuous quantity to one of three qualitative states (increasing, decreasing, or constant). Temporal intervals can then be defined based on these states. For example, dropping an object may be defined in part by the object’s height decreasing over some interval.

4.3 Case Construction

Case construction starts by recording a trace of an event demonstration. Recall that an event demonstration is accompanied by a natural language description of the event (e.g. push the green sphere). For recording a trace, AILEEN attempts to ground language referents to objects in the scene. During a demonstration, a trace is stored for changing quantities between AILEEN and these ground referents. A list of these quantities and

their corresponding encodings is given in Table 1. The trace records the instantaneous value for each quantity every .01 seconds.

Quantity	Encoding
distance(SUBJ, HAND)	Derivative Sign
HEIGHT(SUBJ)	Derivative Sign
distance(SUBJ, AILEEN)	Derivative Sign
POSITION Left/Right(SUBJ)	Derivative Sign
HAND/SUBJ Bounding Boxes	rcc8
moving(HAND)	Moving or Stationary

Table 1: Quantities and encodings

From a trace, a qualitative spatiotemporal representation can be constructed. For each event demonstration, a set of intervals are generated. Figure 1 shows encoded intervals for a 'push' action.

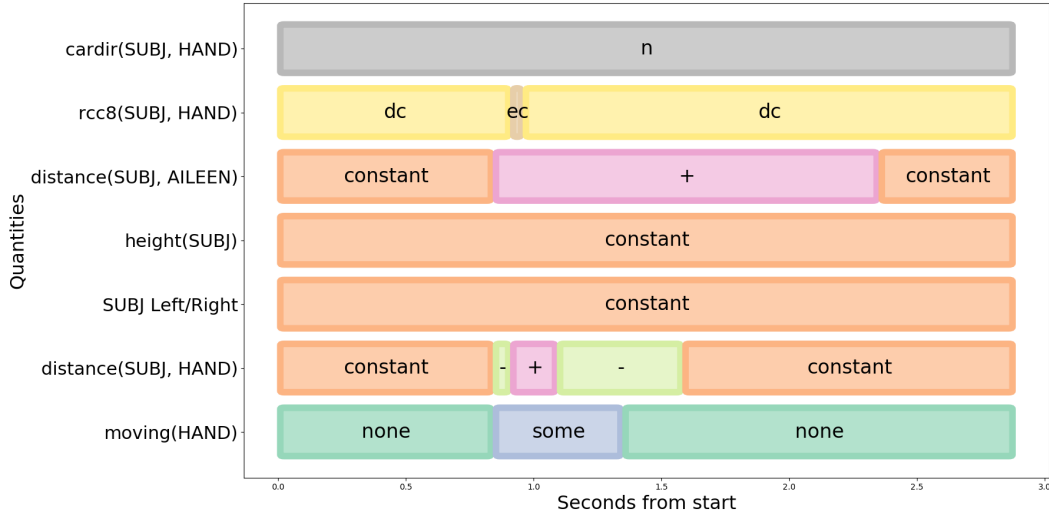


Figure 9: Quantity Intervals for a push example.

In order for learning to occur, these intervals are composed into a declarative graph. Intervals are related to each other using Allen Interval Algebra [1]. Construction begins by creating an interval that spans the entire duration of the event EPISODE_INTERVAL. This is then related to intervals corresponding to the movement of AILEEN's hand moving(HAND). These HAND_MOVING intervals are related to all other temporally local quantity intervals, i.e. where the AIA relation is not *precedes* or *preceded-by*. Figure 9 shows all intervals for a *push* example. Additionally, the temporal *meets* relation is used to indicate a linear progression within each quantity type. For example, *distance(SUBJ, AILEEN)* will result in *meets(constant-interval, +-interval)* and *meets(+interval, constant-interval)*.

4.4 Results

Current results show that AILEEN is able to learn concepts from few examples. Figure ?? shows that the 'drop' concept is learned after three examples, i.e. AILEEN can recognize new positive instances of the concept.

Case Facts	Gen Facts	P
(isa EP1 INSTRUCTION_EPISODE)	(isa (GenEntFn 16 0 r_pushMt) INSTRUCTION_EPISODE)	1.0
(noneIn EP2 HAND_MOVING)	(noneIn (GenEntFn 0 0 r_pushMt) HAND_MOVING)	1.0
(someIn EP3 HAND_MOVING)	(someIn (GenEntFn 1 0 r_pushMt) HAND_MOVING)	1.0
(noneIn EP4 HAND_MOVING)	(noneIn (GenEntFn 2 0 r_pushMt) HAND_MOVING)	1.0
(meets EP2 EP3)	(meets (GenEntFn 0 0 r_pushMt) (GenEntFn 1 0 r_pushMt))	1.0
(meets EP3 EP4)	(meets (GenEntFn 1 0 r_pushMt) (GenEntFn 2 0 r_pushMt))	1.0

Table 2: Example facts and the corresponding concept generalization. The actual generalization has 176 facts so many are omitted.

Specificity drops, however. This indicates that AILEEN recognized one false positive. In other words, there is overlap between a negative example and the learned 'drop' model. We are looking into addressing this by utilizing near-miss learning [3].

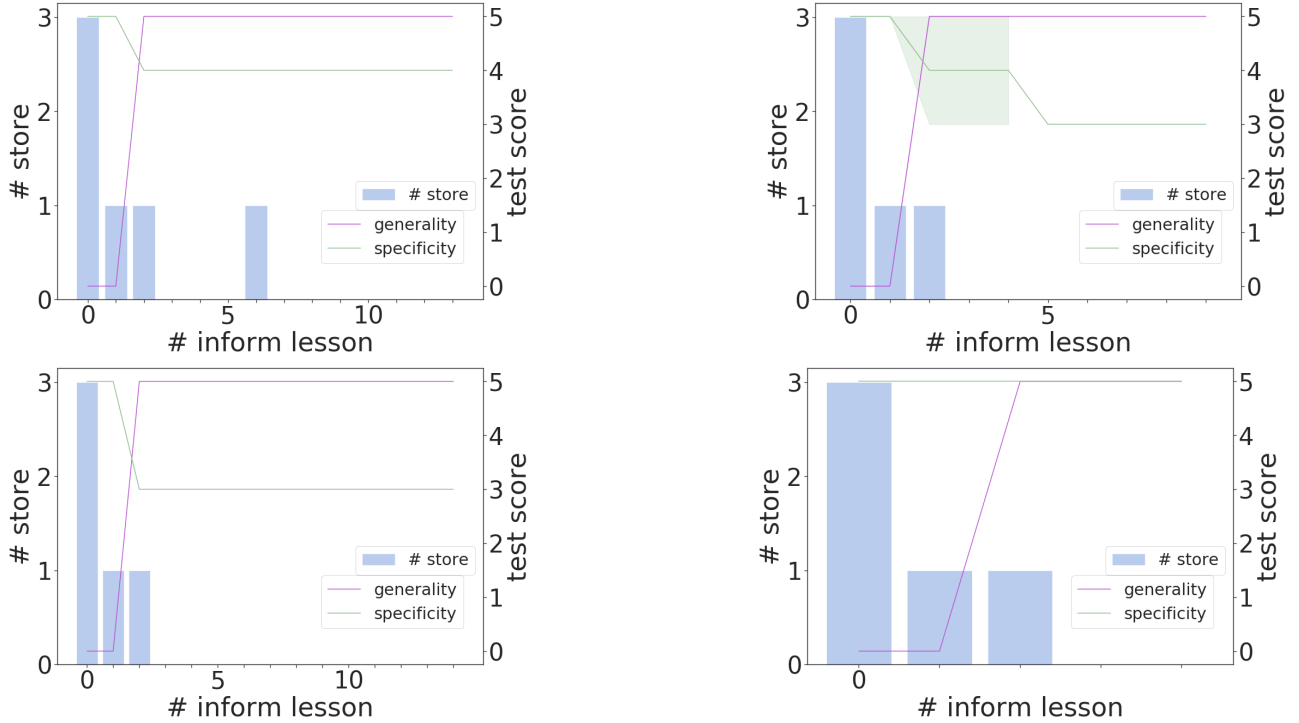
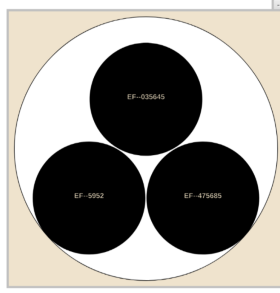


Figure 10: Results for learning the concept *drop*, *push*, *pull*, *pick*

Figure 11 shows the representations learned by AILEEN. Representations used by AILEEN are compositional, explicit, and encode the statistics of the domain. We can see that AILEEN can learn a general representation that when an object is push, the distance between objects reduce. The specifics of demonstrations - that the object was of a certain color are dropped from inference because they are low probability facts.



Three examples assimilated into the 'push' generalization pool

	probability
(aia- (GenEntFn 4 0 r_pushMt) (GenEntFn 12 0 r_pushMt))	1.0
(aia-d (GenEntFn 2 0 r_pushMt) (GenEntFn 9 0 r_pushMt))	1.0
(aia-n (GenEntFn 9 0 r_pushMt) (GenEntFn 12 0 r_pushMt))	1.0
(aia-nl (GenEntFn 6 0 r_pushMt) (GenEntFn 7 0 r_pushMt))	1.0
(changingIn (GenEntFn 9 0 r_pushMt) (PositionFn HAND))	1.0
(constantIn (GenEntFn 15 0 r_pushMt) (DistanceFn SUBJ AILEEN))	1.0
(decreasingIn (GenEntFn 3 0 r_pushMt) (DistanceFn SUBJ HAND))	1.0
(holdsIn (GenEntFn 1 0 r_pushMt) (dc SUBJ HAND))	1.0
(isa (GenEntFn 0 0 r_pushMt) INSTRUCTION EPISODE)	1.0
(r_push (GenEntFn 0 0 r_pushMt))	1.0
(aia- (GenEntFn 5 0 r_pushMt) (GenEntFn 7 0 r_pushMt))	0.6666667
(aia-d (GenEntFn 10 0 r_pushMt) (GenEntFn 9 0 r_pushMt))	0.6666667
(aia-n (GenEntFn 2 0 r_pushMt) (GenEntFn 10 0 r_pushMt))	0.6666667
(holdsIn (GenEntFn 17 0 r_pushMt) (dc SUBJ HAND))	0.6666667
(aia- INT58 (GenEntFn 12 0 r_pushMt))	0.3333334
(aia-d (GenEntFn 7 0 r_pushMt) (GenEntFn 0 0 r_pushMt))	0.3333334
(aia-di (GenEntFn 1 0 r_pushMt) (GenEntFn 7 0 r_pushMt))	0.3333334
(aia-di (GenEntFn 8 0 r_pushMt) (GenEntFn 7 0 r_pushMt))	0.3333334
(aia-di (GenEntFn 10 0 r_pushMt) (GenEntFn 7 0 r_pushMt))	0.3333334
(aia-di (GenEntFn 11 0 r_pushMt) (GenEntFn 7 0 r_pushMt))	0.3333334
(changingIn INT58 (PositionFn HAND))	0.3333334
(holdsIn INT35 (dc SUBJ HAND))	0.3333334
(holdsIn INT58 (ec SUBJ HAND))	0.3333334
(holdsIn (GenEntFn 17 0 r_pushMt) (po SUBJ HAND))	0.3333334

Probability of a fact occurring in a generalized example

Facts that fall under a probability threshold of .5 are ignored in similarity measures

Figure 11: Caption

5 Full AILEEN System

The final architecture developed for Phase III is shown in Figure 12. The green boxes were added during Phase III while the remainder was built during Phase I and II.

References

- [1] James F. Allen. "Towards a general theory of action and time". In: *Artificial Intelligence* 23.2 (1984), pp. 123–154. issn: 0004-3702.
- [2] Yiannis Gatsoulis et al. "Qsrlib: a software library for online acquisition of qualitative spatial relations from video". In: (2016).
- [3] Matthew Mclure, Scott Friedman and Kenneth Forbus. "Extending Analogical Generalization with Near-Misses". In: (12/2017).

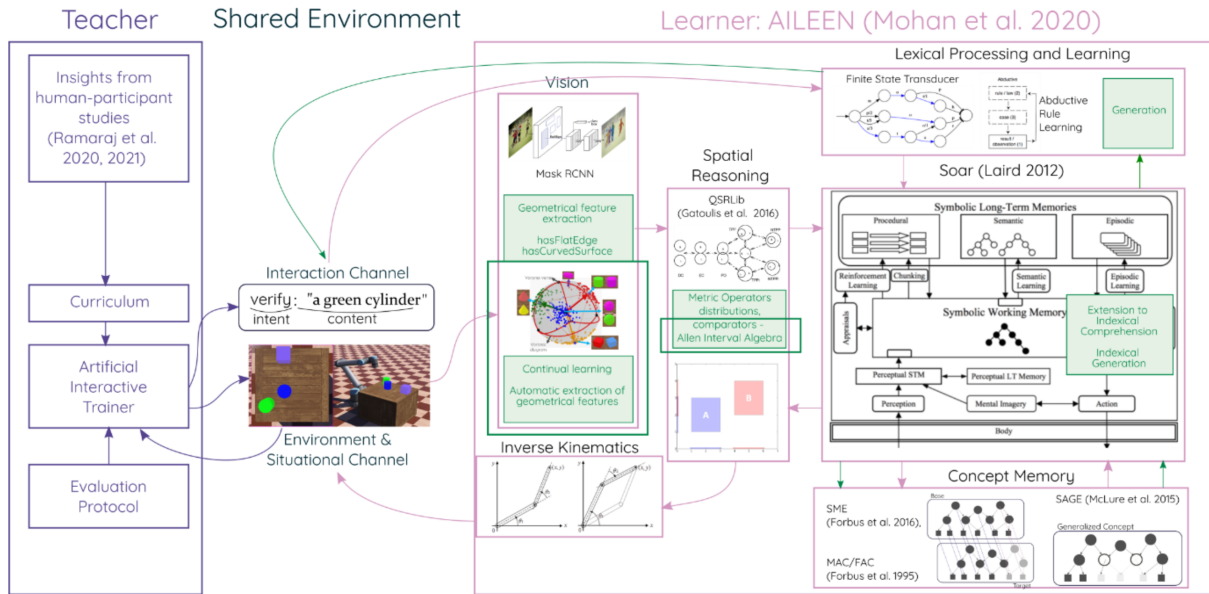


Figure 12: AILEEN architecture