Analogical Generalization of Activities from Single Demonstration

Jason R. Wilson and Matthias Scheutz

Human-Robot Interaction Laboratory
Tufts University
200 Boston Avenue
Medford, MA USA

Abstract. Learning new activities (i.e., sequences of actions possibly involving new objects) from single demonstrations is common for humans and would thus be very desirable for future robots as well. However, "one-shot activity learning" is currently still in its infancy and limited to just recording the observed objects and actions of the human demonstrator. In this paper, we introduce a process called "Mental Elaboration and Generalization by Analogy" to create a generalized representation of an activity that has been demonstrated only once. By abstracting over various dimensions of the learned activity, the obtained activity representation is applicable to a much wider range of objects and actions than would otherwise be possible.

Keywords: learning generalization analogy robot

1 Introduction

Learning new activities on robots from human demonstration has been investigated for quite some time (e.g., see [1] for an overview). However, there are surprisingly few attempts to learn activities in a more natural, human-like way from a *single demonstration paired with natural language instructions* (e.g., see [5]). The challenges in such "one-shot activity learning" include the processing of fairly unconstrained task-based natural language instructions in real-time and the difficulty of integrating demonstrations and natural language instructions in a mutually synergistic way.

Determining the relevant features that constitute the activity is another problem. It is believed that human learners use their imagination invoking a mental simulation to construct variations of the scenario. They are then able to distinguish those features that are relevant to the activity from those that are not.

In this paper, we introduce our first attempts at using mental simulations to learn relevant features of single presentations of activities. Specifically, we describe a first set of algorithms that can generate novel, yet similar, situations from a given situation and use analogical reasoning to determine which of the different features in the new situation matter for the previously observed activity. From these simulations, we then generate a more abstract activity description that includes the most relevant relations and entities.

2 Background

Human lives are filled with learned activities that are performed on a daily basis, from driving to the grocery store, selecting and fetching appropriate items from the shelves, packing them in grocery bags at the cash register, stacking the fridge at home, to preparing meals, cleaning the dishes, and so forth. Humans are particularly good at learning and executing such activities, which are often learned from one single demonstration by a teacher – we will call this "one-shot activity" learning. In *one-shot activity* learning, the teacher typically uses a mixture of natural language instructions, gestures, and action demonstrations to teach the activity, while the learner uses multi-modal cues to make sense of the demonstrated activity. This includes determining the relevant objects and actions, as well as the appropriate sequencing of actions and events. When successful, the learner will have formed an appropriate "activity representation" that removes irrelevant physical details (e.g., the distance of the target object from the hand before a grasp or the particular motion trajectories on the way to the grasp, etc.) while capturing relevant details (e.g., the target object needs to be picked up and placed in a particular location relative to another object).

Recent work in robotics has proposed solutions to different aspects of human activity learning. For example, [2] showed how to learn new (primitive) actions from natural language instructions. Beyond actions, [11] demonstrated how various additional properties of objects could be learned from one-shot natural language descriptions. [12] demonstrated how a robot could learn to follow recipes written in natural language on wikihow.com, also utilizing a variety of corpora (the WordNet lexical database. The KeJia project has also made progress in allowing robots to learn from written natural-language data [3] when gaps in the robot's knowledge base are detected. However, none of these projects allow for learning abstractions from single presentations that lead to generalized activity representations which include novel situations with new objects and features.

3 Architecture

The components and control flow of Mental Elaboration and Generalization by Analogy (MEGA) are shown in the gray box of Figure 1. Also in the figure is the control flow for a new scenario. The Measure Applicability of Novelty (MAN) component receives a new scenario and compares it to the generalization using the "structure mapping engine" (SME) [7]. If the generalized activity, which includes a partial description of the context in which it may be applied, and the new scenario are sufficiently similar, then the activity is applicable to the scenario and analogical inferences generated by SME are inspected to find the appropriate variable bindings. In the nest section, we will describe the two main components of MEGA, Mental Elaboration and Generalization by Analogy.

The components of MEGA and MAN are are designed to be easily integrated into the DIARC architecture [13]. We have defined the interface to the Action Manager component to receive the propositional description of the demonstrated

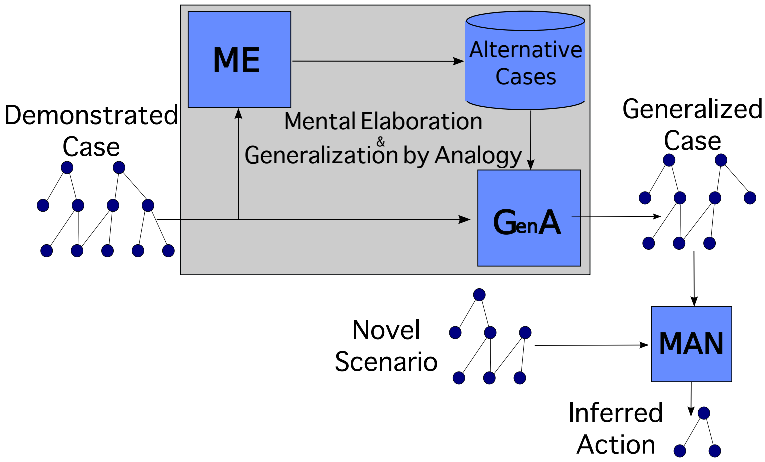

Fig. 1. MEGA process creates a generalized activity, which is later used to infer appropriate activities and variables in a novel situation.

activity. MEGA then internally stores the generalized activity. Later, when the Action Manager is selecting the appropriate actions to complete an activity, it provides the new scenario to the MAN component, which will compare the scenario with the generalized activities and return the applicable action.

4 MEGA

We will tackle the generalization of activities from single demonstrations using SME. Similar to [9], we attempt to find correspondences between different cases (*activities* in our work) using structure mappings, but different from [9] we determine the generalization over all comparisons in parallel (instead of a pair-wise sequence). Moreover, because we are concerned with the generalization of a single activity, all comparisons are made to the demonstrated target activity. Our approach uses a two-phase process. In the first phase (ME in Figure 1) a single demonstration of an activity, e.g., “picking up a medical kit” as might be critical for a search-and-rescue robot – the situation encoded in a “propositional frame” is elaborated upon to develop alternatives cases. Each valid case is then compared to the original case in the second phase (GenA). Comparisons attempt to find an analogical mapping between the cases. Based on these mappings and an evaluation of the similarity between the cases, the most salient items in the case are identified and incorporated into a generalization of the activity.

4.1 Mental Elaboration

The Mental Elaboration (ME) phase of MEGA produces a set of alternative cases based on a single given case. This process is described in Algorithm 1. The case is altered along two dimensions, feature and object, to create combinations

Algorithm 1 Mental Elaboration

Require: $base$ case - propositional frame describing demonstrated action
$H \leftarrow \{base\}$
$H \leftarrow H \cup objAlts(base)$
$OUT \leftarrow \emptyset$
for all $hcase \in H$ **do**
  $candidates \leftarrow featAlts(hcase)$
  for all $c \in candidates$ **do**
    if valid(c) **then**
      $OUT \leftarrow OUT \cup c$
    end if
  end for
end for
return OUT

of variations. We begin with generating the object alternatives (*objAlts*) for the base case, and then generating a set of feature alternatives (*featAlts*) for each object alternative. Together this creates a set of candidate cases that are varied along the feature and object dimensions. Candidates are filtered to check for validity and consistency before being accepted as an alternative case for generalization consideration.

Object dimension. The object dimension represents different entities that are involved in the context on the activity. The intent of the object dimension is to identify the range of objects to which the activity can be applied. The mental elaboration process imagines other objects that may fit the scenario. To illustrate this process, we will use the example of a "medical kit" that is defined as "a white box with a red cross on it and a handle on top". The alternative cases will range over objects that have handles, some of which may be highly similar to the medical kit, such as toolboxes or a suitcase. Some objects, however, have handles in a different orientation, like a mug or a milk jug. Other objects have multiple handles, like a suitcase (one on the side, one on top) and a tote bag (two handles that come together to form a single handle). Additionally, objects that have handles but are not intended for lifting, like a door, are also included as alternative cases. Table 1 lists all considered objects, including the number of alternative cases used in the generalization phase. The total includes variations along the feature dimension and then checked to meet basic validation constraints. Each of these processes are discussed further below.

Each scenario - which includes the object to be lifted, the agent doing the lifting, and descriptions of the start and end states - is captured in a propositional frame called a "case". For each alternative object, a duplicate case is generated with the original object and its corresponding facts removed and the new object and its facts inserted in its place. The facts relating to the object are defined in the knowledge base and include features like the shape, color, and position and location of handle(s). Examples of facts from the knowledge base related to a a medical kit and those related to one alternative object are shown in Table 2.

Table 1. The objects considered vary in the number of handles, location of the handle, and the purpose of the handle. The alternative objects generated that pass the validation constraints are the source of the generalized action.

object	# handles	handle location(s)	lifting handle	alternatives
medical kit	1	top	yes	144
toolbox	1	top	yes	1512
suitcase	2	top/side	yes	5292
basket	1	top	yes	648
tote bag	2	top	yes	648
mug	1	side	yes	324
milk jug	1	side	yes	432
door	1	side	no	432

Feature dimension. Exploring the relevant feature space is done through alterations along the feature dimension. We include amongst the features not only attributes like color and shape but also relations like spatial orientation. The intent of making alterations along this dimension is to identify which features are significant to the activity to be generalized. For example, in most cases the color of an object has no influence on how one manipulates the object. However, the orientation of the handle relative to the object is significant in many activities.

Only a small set of the features may be integral and necessary for the activity, and the challenge is to efficiently identify them. In analogical comparisons, greater contributions to the similarity of two cases originate from higher-order relations and enhance the systematicity of the case. Hence, we expect that the features that are less important contribute less to the overall structure of the representation of the context and should not be included in the generalization of the action. One example of the difference between significant and insignificant features is the color of the medical kit versus the force exerted on the handle of the medical kit. The color is only referenced in relation to the medical kit, and thus the color of the medical kit is irrelevant. The upward force exerted on the handle is referenced in three relations, thus it is a significant feature and is necessary for lifting the object by its handle.

Note that recognizing these differences is problematic for many statistical approaches that primarily rely on frequency of features. E.g., if the majority of the objects lifted are white, then it is inferred that the action requires the object to be white. Similarly, to teach a robot to place a yellow object on a higher platform requires a series of trials consistently demonstrating the proper placing of the yellow object [6]. The intent of our approach is to leverage mental simulation and analogy-based comparisons to enable the identification of the significant and influential features of the scenario from a single demonstration. If the color of the object is significant to the action, then this is captured in the representation of the scenario with the color being related to some other entity in the scenario. This information may originate from natural language instruction and then encoded as a proposition in the scenario.

Table 2. The facts on the left are related to a medical kit and are replaced by the facts about a mug on the right when generating the alternative case for the mug.

medical kit	mug

`(medical-kit ?instance (`
`(medical-kit ?instance)`
`(box ?instance)`
`(rigid-object ?instance)`
`(has-color ?instance white1)`
`(white white1)`
`(greater-than (width ?instance)`
`(height ?instance))`
`(handle handle0)`
`(graspable handle0)`
`(has-part ?instance handle0)`
`(on-top-of handle0 ?instance)`
`(connection c1)`
`(end-of c1 ?instance)`
`(end-of c1 handle0)`
`(rigid-object c1)`
`(left-of c1`
`(center-of-mass ?instance))`
`(connection c2)`
`(end-of c2 ?instance)`
`(end-of c2 handle0)`
`(rigid-object c2)`
`(right-of c2`
`(center-of-mass ?instance))))`
	`(mug ?instance (`
	`(mug ?instance)`
	`(cylinder ?instance)`
	`(rigid-object ?instance)`
	`(has-color ?instance white1)`
	`(white white1)`
	`(equal (width ?instance)`
	`(height ?instance))`
	`(handle handle0)`
	`(graspable handle0)`
	`(has-part ?instance handle0)`
	`(on-side-of handle0 ?instance)`
	`(connection c1)`
	`(end-of c1 ?instance)`
	`(end-of c1 handle0)`
	`(rigid-object c1)`
	`(right-of c1`
	`(center-of-mass ?instance))`
	`(connection c2)`
	`(end-of c2 ?instance)`
	`(end-of c2 handle0)`
	`(rigid-object c2)`
	`(right-of c2`
	`(center-of-mass ?instance))))`

Modifying propositions and creating sets of modifications generates new alternative cases. Changes along the feature dimension consists of modifying classes, attributes, and relations. Table 3 lists the features of each type that are included in the demonstrated scenario. The knowledge base contains a range of possible values for each feature.

Each type of feature requires its own rules for finding alternative values. Each rule follows the pattern of verifying that the proposition is true in the scenario, identifies a value for that feature, and verifies that the new value is different than the original. A rule for changing the class of an entity is the following:

```
(<- (change-class (?type ?entity) (?new-type ?entity))
    (instanceof ?entity ?type)   ;; entity is an instance of type
    (isa ?type ?super)           ;; type is known to have a super class
    (isa ?new-type ?super)       ;; new-type is a member of the class
    (different ?type ?new-type)) ;; new-type and type are not the same
```

The *change-class* rule is applied to propositions in the case that define the class of an entity. Similar rules are used to change attribute values and relationship types. Combinations of these values produces the new cases along the

Table 3. Each feature and its set of valid values.

type	feature	values
class	agent	human, robot
class	surface	table, floor, shelf
class	color	white, red, blue, gray, maroon, pink, yellow
attribute	shape	box, cylinder, cube, flat, spherical
relationship	spatial	left-of, right-of
relationship	inequality	equal, less-than, greater-than

feature dimension. For example, an alternative case for lifting the medical kit will have the agent modified to be a robot. Another case will have the robot as the agent and the color of the kit is yellow.

Validation of features. Not all feature values or combinations of values are always valid for all objects (e.g., cubic or box-shaped mugs are invalid). The validation rules, which are based on knowledge of valid features of an object, verify that the combination of features is acceptable. For example, we require that any medical kit be either red or white. If no validation rules pass, then the case is rejected. Once a generated case has been determined to be valid, it is made available for the generalization process. The tally of valid cases generated for each object type is shown in Table 1.

MEGA requires a knowledge base of facts and rules that are assumed background knowledge needed to quickly learn. However, the whole MEGA process requires very little knowledge. The generation and validation of the alternative cases requires less than 250 facts and 50 rules. The majority of the facts are descriptions of objects, and the most of the rules define the Mental Elaboration (ME) process and are independent of the objects and features included in the knowledge base. Expanding the knowledge base to incorporate additional objects to be used in the ME process mostly requires an addition of roughly 20 facts (see examples in Table 2). Constraints on valid features of each object is currently defined as a set of rules, which also would need to be added to the knowledge base when including new objects.

4.2 Generalization by Analogy

Once all the alternative cases have been generated, we identify the similarities in the cases to construct the generalization. Using a computational model of analogy, we can select the items in the cases that are most significant. The structure mapping engine (SME) [7] is an analogical reasoning engine that adheres to the principles of structure mapping theory [8]. SME compares representations of the two cases by finding correspondences between items in each representation. This processes is governed by the constraints of *structural consistency*: *one-to-one mapping* and *parallel connectivity.* A one-to-one mapping means that a structurally consistent mapping between the base case and the target case does not include any item in the base case being mapped to more than one item in the target case, and vice versa. Parallel connectivity requires the arguments of a pair

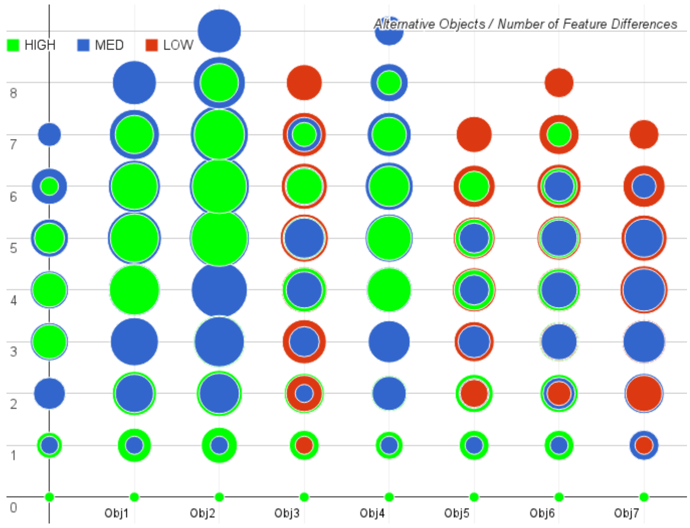

Fig. 2. The horizontal axis has the alternative objects considered, from left to right: medical kit, toolbox, suitcase, basket, tote bag, mug, milk jug, and door. The vertical axis is the number of feature differences. The size of the circle is how many cases were generated. The color of the circle shows how similar the case is to the original case of lifting a medical kit.

of statements to be mapped if the statements are in correspondence. Additionally, SME implements the *systematicity principle*, which requires relations that are included in the analogy to be part of a system of relations. It has been shown that people identify the key facts in an analogy as the one that are part of a system of connected relations [4].

These principles are key in identifying salient elements in a scenario where an action is used. For example, a one-to-one mapping associates the agent in the base case doing the lifting to the agent in the new case doing the lifting. Mapping it to multiple agents is not appropriate as a two agent lifting action has many differences compared to a single agent lifting. Systematicity is also crucial because it helps differentiate between elements that are essential in the scenario and those that are just surface features. Essential elements are likely to take part in multiple relations. For example, the hand is essential in the lifting action. It is part of the relations defining it as part of the agent, as part of the action, and as the thing grasping the object. It is not necessary that it be a hand, but some common entity must fill the role of being part of the agent, be involved in the lifting action, and grasping the object.

The analogical comparison process done by SME constructs proposed correspondences between an item in the base case and an item in the target. Each correspondence, called a *match hypothesis*, is assigned a score based on an initial value and some value inherited from a parent match hypotheses. The sum of all the match hypothesis scores produces the overall structural evaluation score (SES) for the whole comparison. This score is a measure of the structural soundness of the comparison and is thus a measure of the strength of the analogy. A poor analogy, or a comparison that has few relations in common, will have fewer match hypotheses with relations and thus less scores that can trickle down and accumulate in match hypotheses containing the arguments of the relation.

Our approach to generalization by analogy takes advantages of the structural evaluation score and the match hypothesis score. The base case is compared to each target case generated by ME. Each item in the base case is evaluated for its significance to the case. For each item i in the base case, the significance score (Sig) is the weighted sum of the match hypothesis score (MHS) in which it occurs in each case compared to the base. The weight applied is the structural evaluation score (SES) for the base/target comparison.

$$Sig_i = \sum_{c \in cases} SES(base, c) * MHS(i, base, c) \tag{1}$$

Extending this further, we can compare the base case to all of the variations that were automatically generated as part of the Mental Elaboration process. Figure 2 shows the SES of each of these comparisons. We now introduce GenA, an engine for automatically doing these comparisons, identifying the most significant elements of the cases, and producing a generalization of the base case. Algorithm 2 describes this process. It begins with the demonstrated cases and the alternative cases generated from it. For each of these generated cases, it is compared with the demonstrated case. This is an analogical comparison that produces a set of mappings between items in the cases. The mapping with the best score is further analyzed. For each match hypotheses (*mh*) in the mapping, the score of the *mh* is added to the score of the base item in the *mh*. At this point, each item has a a significance rating associated to it. If the item significance is greater than a threshold, it is included in the generalization.

Filtering all the expressions to include only those with a significance greater than some threshold ensures that the generalization includes only the most important expressions. If an expression has a valid mapping in each analogy then it is likely the expression is necessary for the generalization. If it has a mapping in the majority of the cases, then it is still likely to be an important expression. Given these assumptions, we calculate the threshold as the minimum score for an expression if it has a mapping in the majority of cases and each of these cases is a perfect analogy.

5 Measure Applicability of Novelty

When a new, and potentially novel, scenario is presented, a comparison of the scenario with the generalized action case reveals the applicability of the action to the new scenario. This process, called Measure Applicability of Novelty (MAN), uses SME to compare the new case to the generalized case. If the cases are sufficiently similar, then the inferences produced by SME are examined. The candidate inferences provides potential projections of the generalized case onto the novel case. The most important projections are the action and its parameters.

Algorithm 2 Generalization by Analogy

Require: *base* - demonstrated case
Require: *generatedCases* - generated alternative cases
  $items \leftarrow items(base)$
  for all $gencase \in generatedCases$ **do**
    $mappings \leftarrow$compare($base, gencase$)
    $bm \leftarrow$greatestSES($mappings$)
    $ses \leftarrow$score(bm)
    for all $mh \in$mhs(bm)) **do**
      $baseItem \leftarrow$base(mh)
      if $baseItem \in items$ **then**
        sig($baseItem$) $\leftarrow$score($baseItem$) + score(mh)
      end if
    end for
  end for
  $threshold \leftarrow$computeThreshold($base$)
  $keeprItems \leftarrow$filter($items, threshold$)
  return expressions($keeprItems$)

The comparison with the generalization and inferences made about the new situation is a similar process to that used in case-based reasoning [10]. However, SME is a domain-general engine for adapting the new scenario. Additionally, it provides a common mechanism across the MEGA and MAN components.

We evaluate the quality and utility of the generalized action by presenting a novel scenario to MAN. It is our hypothesis that the generalized action will be most similar to novel scenarios that have a handle for lifting on top of the object. It is also expected that the process will be resilient to feature variations that were not previously seen.

The novel scenario has a brown briefcase sitting on a desk. The goal is for the robot to be holding the briefcase above the desk. We also introduce additional features to the scenario to show that these are easily ignored. We add that the handle is black and, the briefcase is metallic, and the briefcase is heavy. The intent is for MAN to determine that the generalized action case for lift-up is similar to this novel scenario and that the parameters to the lift-up activity are the robot and its hand.

Comparing the generalized activity to a variety of other novel scenarios shows a consistent pattern. In all these novel cases, the surface is a desk, which was never seen in the original demonstration or the mental elaboration of it. We also introduce new objects: a hammer, a desk drawer, a cooking pan, and a soda can. It is important to note that a soda can clearly does not have a handle, and it would be anticipated that this scenario would be less similar. Additionally, we try objects that were included in the elaboration phase but with never seen before features (e.g., black handles, red cross on medical kit, and suitcase with a single handle). In keeping consistency with our intent to have the robot learn to perform these actions, all of these novel cases have the robot performing the

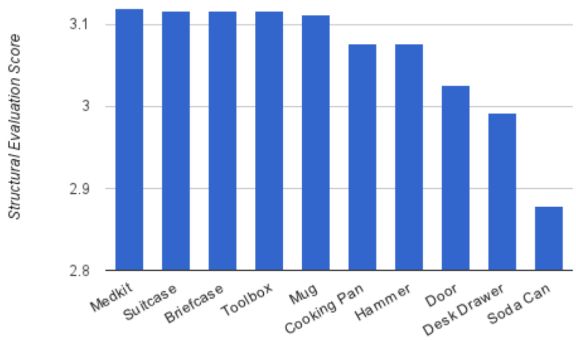

Fig. 3. The generalized action is most similar to objects with a handle on top. Other orientations and uses of handles result in even less similarity. Finally, an object without a handle is least similar.

action. Figure 3 shows that one class of scenarios (medical kit, suitcase, briefcase, toolbox, and mug) have nearly the same similarity score to the generalized case. This similarity drops off as the action becomes less applicable to the scenario. The cooking pan and hammer each have a handle, but the handle is attached to the object in a different manner, leading to the diminished similarity. The door and the desk drawer have handles, but they are not lifting handles. Finally, the least similar case is a soda can because it does not have a handle.

6 Discussion and Conclusion

Being able to quickly learn is not only a feature of human learning but is critical for developing natural interactions with robots. We have presented a system that learns to apply an activity to novel situations based on a single demonstration of the activity. Through a process of Mental Elaboration and Generalization by Analogy (MEGA), we are able to determine the most significant elements in the application of an action and create a generalized action. We demonstrated the quality of the generalization by comparing to new situations. In addition to the comparison results being as expected, the analogical comparison produces inferences that includes the action and the proper bindings of the action's variables.

While the solution proposed here bears great promise, there are some potential issues. The representation of the cases is important, but much of the information included is general world knowledge. The Mental Elaboration phase relies on a knowledge base that contains facts about the structure and features of objects and ontological relations of entities. However, rapid learning by humans is also greatly dependent on background knowledge. In addition to requiring a relatively small knowledge base, SME (which is at the core of our approach) does not require any background knowledge.

Future work will focus on integrating the algorithms introduced in this paper into a robotic architecture. This type of integration will bring various challenges such as extracting the symbolic expressions needed for MEGA from the perceptual components. However, it will also allow us to utilize other features of the architecture, namely the simulation components. Supplementing MEGA with a simulation of cases that are sufficiently similar will reveal which of the cases are physically impossible and thus can be ruled out. An additional future development is reducing the overall computational complexity of MEGA. The greatest efficiencies can be gained from limiting object alternatives to similar objects, a heuristic search on which features to vary, and only generating enough alternative cases to construct a cohesive generalization.

Acknowledgement

This work was supported by ONR BRC #N00014-11-1-0493.

References

1. B. D. Argall, S. Chernova, M. Veloso, and B. Browning. A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57:469–483, 2009.
2. R. Cantrell, P. Schermerhorn, and M. Scheutz. Learning actions from human-robot dialogues. In *Proc. 2011 IEEE Symp. Robot and Human Interactive Comm.*, 2011.
3. X. Chen, J. Xie, J. Ji, and Z. Sui. Toward open knowledge enabling for human-robot interaction. *J. of Human-Robot Interaction*, 1(2):100–117, 2012.
4. C. Clement and D. Gentner. Systematicity as a selection constraint in analogical mapping. *Cognitive Science*, 132:89–132, 1991.
5. P. F. Dominey, A. Mallet, and E. Yoshida. Real-time spoken-language programming for cooperative interaction with a humanoid apprentice. *Intl. J. of Humanoid Robotics*, 6(2):147–171, 2009.
6. W. Erlhagen, A. Mukovskiy, E. Bicho, and G. Panin. Goal-directed Imitation for Robots : a Bio-inspired Approach to Action Understanding and Skill Learning. *Robotics and autonomous systems*, 54(5):353–360, 2006.
7. B. Falkenhainer, K. Forbus, and D. Gentner. The structure-mapping engine: Algorithm and examples. *Artificial Intelligence*, 41:1–63, 1989.
8. D. Gentner. Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7:155–170, 1983.
9. S. Kuehne, K. Forbus, D. Gentner, and B. Quinn. Category learning as progressive abstraction using structure mapping. In *Proc. 22nd Cognitive Sci. Conf.*, 2000.
10. R. Lopez De Mantaras, D. McSherry, D. Bridge, D. Leake, B. Smyth, S. Craw, B. Faltings, M. L. Maher, M. T. Cox, K. Forbus, et al. Retrieval, reuse, revision and retention in case-based reasoning. *Knowledge Eng. Review*, 20(03):215–240, 2005.
11. S. Mohan, A. Mininger, J. Kirk, and J. E. Laird. Learning grounded language through situated interactive instruction. In *AAAI Fall Symp. Series*, pages 30–37, 2012.
12. D. Nyga and M. Beetz. Everything robots always wanted to know about housework (but were afraid to ask). In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ Intl. Conf. on*, pages 243–250. IEEE, 2012.
13. M. Scheutz, P. Schermerhorn, J. Kramer, and D. Anderson. First steps toward natural human-like HRI. *Autonomous Robots*, 22(4):411–423, May 2007.