Situated Bayesian Reasoning Framework for Robots Operating in Diverse Everyday Environments

Sonia Chernova, Vivian Chu, Angel Daruna, Haley Garrison, Meera Hahn, Priyanka Khante, Weiyu Liu, Andrea Thomaz

Abstract General-purpose robots operating in unstructured environments have the potential to benefit by leveraging abstract, commonsense knowledge for task execution. In this paper, we present an approach for automatically generating a compact semantic knowledge base, relevant to a robot's particular operating environment, given only a small number of object labels obtained from object recognition or a robot's task description. In order to cope with noise and non-deterministic data across our data sources, we formulate our representation as a statistical relational model represented as a Baysian Logic Network. We validate our approach in both abstract and real-world domains, demonstrating the robot's ability to perform inference about object categories, locations and properties given a small amount of local information. Additionally, we present an approach for interactively validating the mined information with the help of a co-located user.

1 Introduction

When a robot is tasked to operate in a new environment, it should have the ability to leverage external knowledge sources to acquire common knowledge about its general environment instead of learning everything from scratch. For example, a maintenance robot should have the ability to leverage common knowledge about tools, just as a home robot should have access to knowledge about household items. Multiple knowledge bases and semantic knowledge graphs have been developed in the AI community that incorporate general, commonsense knowledge; examples include WordNet [13], ConceptNet [18], and ResearchCyc [10, 12], as well as extensions of this work in other research communities, such as ImageNet in computer

Sonia Chernova, Vivian Chu, Angel Daruna, Haley Garrison, Meera Hahn, Weiyu Liu
Institute for Robotics and Intelligent Machines, Georgia Institute of Technology, Atlanta, GA., e-mail: {chernova,vchu,adaruna3,hgarrison3,meerahahn,wliu88}@gatech.edu

Priyanka Khante, Andrea Thomaz
Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX., e-mail: {priyanka.khante,athomaz}@utexas.edu

vision [4]. Prior work has shown that these language-based knowledge resources can be used as a foundation for powerful reasoning methods [1, 24].

Within robotics, several projects have focused on constructing large-scale semantic reasoning databases, often by mining the above resources. The RoboEarth [22] and KnowRob [20] projects have contributed representations that allow semantic data obtained from human labels and ontologies to be stored within the scope of larger robotics architectures. The RoboBrain project [16] seeks to create a massive general-purpose robot knowledge graph encompassing perception, planning, control, natural language and structured knowledge data. Our work considers semantic data mining from a different perspective – given only information available in its environment, how can a robot leverage existing semantic resources to construct a small, situated knowledge base containing semantic information that is both general and uniquely adapted to its particular environment. Thus, instead of creating another general-purpose knowledge base, we study the extent to which local observations can be leveraged to retrieve relevant semantic information at a scale that is more efficient, and often more computationally tractable.

In this paper, we contribute a domain-independent framework for generating a context-specific knowledge network for commonsense reasoning. Given only a set of *seed words*, obtained from object labels from object recognition or a task description (e.g., *fork*, *dishwasher*), our approach leverages existing semantic knowledge bases to construct a unified probabilistic representation that allows for effective inference and generalization over a wide range of tasks (e.g., *IsA(fork,utensil)*, *UsedFor(dishwasher,wash_dishes)*). In order to cope with noise and non-deterministic data across our data sources, we formulate our representation as a statistical relational model that enables efficient forward and reverse inference over the set of known objects. Specifically, we utilize a Bayesian Logic Network (BLN) [8], which combines a set of directed relations between abstract concepts, in our application *IsA*, *AtLocation*, *HasProperty*, and *UsedFor*, with a corresponding probability distribution that models the uncertainty inherent in these relations. We describe how a BLN can be automatically mined from a small number of seed words, and validate its inference performance on 13 abstract domains seeded from object recognition and robot task plans. We then embed the Bayesian reasoning framework within a robot architecture and demonstrate its use in enabling a mobile robot to perform a series of real-world tasks. Finally, we conclude the paper by describing an approach for interactively validating and refining the information stored in the BLN with the help of a co-present human.

2 Related Work

Numerous projects across the AI community have sought to make use of commonsense and semantic knowledge. Three large-scale commonsense knowledge networks used across a wide range of applications are WordNet [13], ConceptNet [18], and ResearchCyc [10, 12]. WordNet consists of a collection of synsets, which connect concepts hierarchically through the *IsA* relation. WordNet also distinguishes between different senses of the same word and provides glosses, or definitions, for

each sense. While WordNet is clean and hand-coded, it also lacks diversity in the types of relations it contains. ConceptNet, on the other hand, contains several dozen different relations, but it does not distinguish between word senses and is largely crowdsourced, leading to a large amount of noise. ResearchCyc uses an even larger number of relations (currently around 17,000) to connect concepts. For the purposes of this work, we choose to use data from WordNet and ConceptNet to take advantage of the complimentary strengths of each. We do not currently use ResearchCyc because its relational structure introduces significant challenges to automatic data retrieval and generalization; prior work leveraging ResearchCyc has largely relied on hand-picked data [20].

In other work, Zhu, et al. [24] perform affordance prediction on a set of images by using a Markov Logic Network (MLN) [15] to represent affordance knowledge. This work also does not deal with context and used hand-selected objects and affordances in the network. In [3], contextual noise is addressed by disambiguating the concepts in ConceptNet to enrich the WordNet senses with more diverse knowledge for improved performance on word sense disambiguation tasks. While disambiguating ConceptNet helped provide context for each of its concepts, the resulting knowledge base contained only abstract information. In contrast to this approach, [19] did construct a situated knowledge hierarchy in a (nearly) automated way, however, the resulting model only included hypernims (the *IsA* relation).

Within robotics, the KnowRob [20] and RoboBrain [16] projects are most closely related to our work. In KnowRob, the authors create a knowledge network from a variety of encyclopedic sources and represented the network using Prolog rules and the Web Ontology Language. This network is then used to repair robot task plans by filling in missing low-level details from high-level task descriptions. However, the KnowRob representation results in a large network without contextual refinement, and the concepts represented therein were manually selected according to perceived relevance to robotic applications rather than automatically generated. In RoboBrain, the authors generate a multimodal knowledge network for robotics using data collected automatically from the web. Similar to KnowRob, the resulting network is very abstract and does not account for the domain-specific details relevant to the situational context of the robot. Finally, the RoboEarth project focused on the creation of a cloud repository of generalizable robot knowledge, including object models and robot task descriptions, that could be transferred across robot platforms and domains [22]. Our work is complimentary to the above research efforts, but differs in its focus, examining instead how local observations can be leveraged to retrieve relevant, domain-specific semantic information.

3 Semantic Knowledge Mining and Representation

A Bayesian Logic Network [8] is a directed statistical relational model in which the variables under consideration are represented as first-order terms or predicates with arguments. BLNs allow logical constraints, represented as first-order logic rules, to be imposed on the network. Prior work in computer vision has utilized Markov

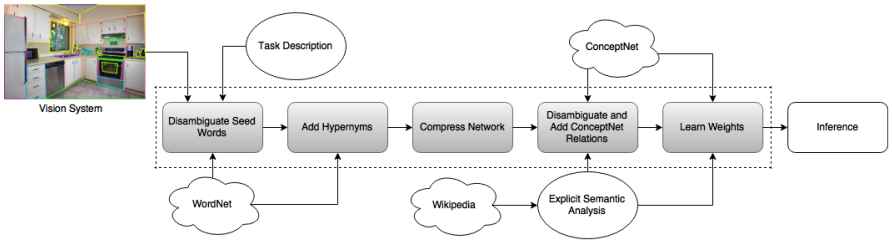


Fig. 1 A flowchart showing the overall approach taken to generate the network. The dashed box shows the components of the algorithm, while white boxes are from external sources.

Logic Networks [15], a representation that unifies Markov Random Fields and first-order logic, for modeling object attributes and affordances [24]. However, parameter learning in MLNs is an ill-posed problem [7] and approximate inference is expensive even for simple queries. In contrast, BLNs are easy to train, more efficient and have scaled better to our application. Figure 3 shows a small example BLN, which, once constructed, can be used to perform inference using likelihood weighting [5] to answer queries such as $AtLocation(Object_i, x)$ or $HasProperty(Object_i, x)$. To construct the BLN, we autonomously mine two online sources of semantic knowledge, WordNet [13] and ConceptNet [18]. Figure 1 presents an overview of the BLN generation pipeline, with main components shown in gray.

Given seed words obtained from object recognition labels, the first step in the pipeline is to perform **word sense disambiguation** to determine the contextually correct senses of the words. Since WordNet provides information on the different word senses, we use it to perform this disambiguation. For example, the word *pan* has the following four senses in WordNet:

1. pan, cooking pan – cooking utensil consisting of wide metal vessel
2. Pan, goat god – (Greek mythology) god of fields and woods and shepherds and flocks
3. pan – shallow container made of metal
4. Pan, genus Pan – chimpanzees; more closely related to Australopithecus than to other pongids

Given a particular environment, not all of the above senses will be contextually relevant. To keep the size of the network small and contextually accurate, we disambiguate the seed words and exclude the irrelevant senses from the generated network. Our approach is similar to that in [21]. Given that the seed words originate from the same context, they are likely to be semantically similar. Therefore, we perform disambiguation by finding the sense of each word that maximizes the overall similarity between the seed words. To do so, the disambiguation algorithm finds a Minimum Spanning Tree (MST), where each node represents the most relevant sense of one of the seed words.

After disambiguating the seed words, we next populate the initial structure of the BLN using **hypernyms**, object categories defined by the *IsA* relation. Categorical

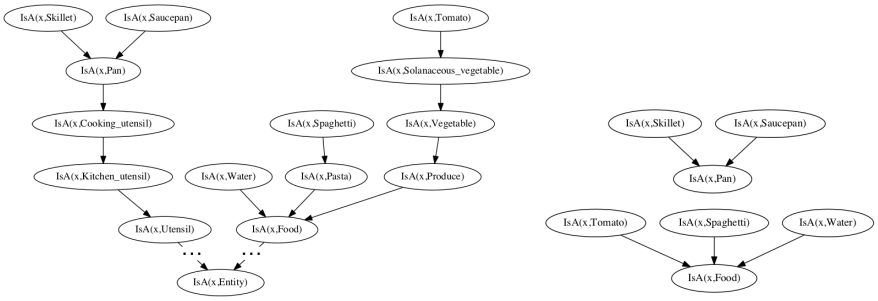


Fig. 2 An example of the *IsA* relation before (left) and after (right) compression. Compression reduces the amount of redundant and high-level information in the network.

information is obtained by traversing the WordNet hypernym hierarchy from each of the disambiguated seed words to the root node in WordNet and adding each node along this path to the network (e.g., *Tomato* $\rightarrow$ *Solanaceous Vegetable* $\rightarrow$ *Vegetable*). Although WordNet is hand-coded, it does contain a large number of redundant and high-level concepts that convey little information, as shown in Figure 2. Including these nodes can lead to rapid expansion in the size of the network, as well as inclusion of uninformative information (e.g., *IsA(Food,Entity)*). To reduce the size of the network to a manageable level and remove the high-level and redundant nodes, we employ a **compression** strategy similar to [19]. The compression uses the following three rules:

1. Eliminate selected top-level, very general, categories (e.g., abstraction, entity).
2. Starting from the leaves, eliminate a parent that has fewer than n children, unless the parent is the root.
3. Eliminate a child whose name appears within the parent's (e.g., Solanaceous Vegetable $\rightarrow$ Vegetable).

For the first rule, we define "top-level" categories as words with an information content [17] of less than 5.0 when evaluated against the Brown corpus [9]. Additionally, we choose $n = 1$ to remove nodes with only a single child. Figure 2(right) shows the example BLN following compression.

Next, we expand the BLN to include the *UsedFor*, *HasProperty*, and *AtLocation* relations from ConceptNet. To do so, we first disambiguate the **relations in ConceptNet** to remove contextually irrelevant relations. ConceptNet does not include sense information, so we use an approach similar to that in [3]. For each ConceptNet relation, $< c, relation, d >$, where d is an ambiguous word and c is disambiguated, we generate the Word Sense Profile, $WSP(d_i) = w_1, w_2, ...$ for each sense, d_i, of the word d. Each w_j in the WSP is a word from one of the following sources in WordNet:

1. All synonyms of d_i
2. All words (excluding stop words) in the gloss/definition for d_i
3. All direct hypernyms (parent nodes) and hyponyms (child nodes) of d_i in WordNet

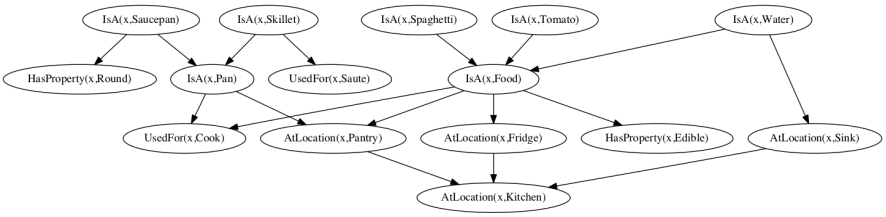


Fig. 3 An example of high-level knowledge representation using a BLN.

4. All meronyms/holonyms (has part or part of) relations in WordNet
5. All words (excluding stop words) in the glosses of the direct hyponyms of d_i

Once we generate the WSP for each sense, we compute a score for each WSP equal to the sum of the semantic relatedness between the non-ambiguous word, c, and each word in $WSP(d_i)$. The relatedness measure is calculated using Explicit Semantic Analysis (ESA) [6] based on word frequency count of Wikipedia entries. We choose the sense of d_i that corresponds with the maximal relatedness value, and add to the BLN only the relations corresponding to the selected senses of each word to avoid contextually irrelevant information from being added to the network. Additionally, we exclude relations, $< c, relation, d >$, where d consists of more than one word. Since ConceptNet is not hand-coded, it contains a significant amount of noisy or erroneous relations, and excluding multi-word relations significantly reduced the size of the network without removing a large number of correct relations.

The final step in BLN construction is to **learn** the conditional probability function for each fragment in the network. To do so, we generate training evidence with a likelihood equal to a linear combination of the weights assigned to each relation in ConceptNet and the Explicit Semantic Analysis relatedness measure between the two concepts in the relation. This approach provides an initial estimate for real-world probabilities and enable inference results to be ranked according to their relative likelihoods. Once we collect the evidence, the conditional probability functions can be learned via maximum likelihood by counting the frequency of each child node being true for each configuration of the parent nodes.

An example resulting BLN is shown in Figure 3. We chose the *IsA*, *HasProperty*, *AtLocation*, and *UsedFor* relations because together they support a wide range of inference queries that can be used across many domain-independent applications. Understanding of object categorization, properties and uses can enable a robot to identify objects that can perform a certain function, or act as a substitute for a missing item. Information about likely object locations can be used in locating objects or act as a prior for semantic mapping. Similarly to the way in which humans use prior knowledge when visiting a new location, we view the presented approach as a means of providing a situated prior to the robot. In this work, we examine the performance of the fixed prior; in future work, we will explore how the resulting model can be refined over time to match a given environment.

SUN1 (20) SUN2 (38) SUN3 (17) ... SUN10 (35) SUN10 labels

Fig. 4 Four of the ten images of kitchens taken from the SUN database, and an example of the accompanying object label information. Number of seed words generated per image is listed in parenthesis.

4 Evaluation of Semantic Inference in Abstract Domains

The knowledge framework described so far has the ability to mine abstract semantic information, but does not yet integrate data from situated robot interactions with the environment. In this section, we briefly evaluate the quality of the abstract data mined, before discussing application and refinement of this knowledge in a robotic setting. We evaluate the performance of our approach on 13 abstract domains using seed words obtained from two different sources:

- Ten segmented and labeled images of kitchen scenes obtained from the SUN database [23] as an analog for object recognition output from a robot's vision system. Figure 4 shows each of the images with the number of seed words from each image. Example seed words obtained for image SUN10 include *basket, cabinet, ceiling, ceiling lamp, clock, coffee maker, cup, curtain, desk lamp, etc*.
- Three task descriptions related to typical household chores, including cooking a recipe (spaghetti), doing laundry, and cleaning the house. Each of the manipulable objects in the three tasks was used as a seed word. For example, the recipe task included objects such as *spaghetti*, *tomato sauce*, and *saucepan*, doing laundry included *shirt*, *laundry detergent*, and *washer*, and cleaning included *vacuum*, *mop*, and *soap*. The three tasks contained 19, 15, and 11 seed words, respectively.

The size of the BLN generated from each of these data sets varied from 69 (cleaning house) to 195 (SUN10) predicates, with an average size of 129.

For each set of seed words, we tested the word sense disambiguation accuracy by computing the percentage of correctly disambiguated seed words. The mean disambiguation accuracy for the three task descriptions was $78.5\% \pm 3.5$, while the SUN images averaged at $83.9\% \pm 6.8$. Overall, the disambiguation algorithm performs better with larger numbers of seed words because the algorithm finds the set of senses that maximizes the similarity between the seed words. This is reflected in the three task description experiments, which had at most 19 seed words, compared to as many as 38 seed words in SUN images. Thus, in a real-world setting, word sense disambiguation will improve as the robot observes more objects.

Next, we evaluate the forward and reverse inference accuracy of the model resulting from each independent set of seed words by comparing the BLN output to a gold standard[1]. For forward inference, we perform inference over the network where the

[1] The gold standard was generated by hand based on commonsense information (e.g., *UsedFor(Knife,Cut)* is `true`), and then validated by comparing to crowd-generated labels from five

Table 1 Forward and reverse (in parenthesis) inference accuracy (%) over the seed words from each source when compared to the gold standard.

Source	IsA	AtLocation		HasProperty		UsedFor	
Recipe	97.6	86.8	(71.4)	82.0	(84.4)	88.1	(88.0)
Laundry	98.3	77.3	(82.1)	88.9	(87.5)	89.5	(90.1)
Cleaning	98.6	72.7	(67.1)	94.7	(90.9)	79.2	(80.0)
SUN1	95.0	81.1	(93.9)	78.0	(85.3)	91.3	(93.5)
SUN2	94.1	84.4	(85.7)	90.4	(91.6)	93.7	(96.3)
SUN3	94.5	76.5	(85.9)	72.4	(81.5)	80.3	(86.9)
SUN4	95.1	78.7	(88.9)	65.2	(78.8)	87.7	(91.7)
SUN5	97.4	83.8	(89.3)	85.0	(87.3)	90.8	(93.7)
SUN6	96.6	73.2	(80.4)	87.2	(90.6)	88.8	(93.6)
SUN7	93.5	74.0	(81.0)	83.6	(90.8)	85.5	(92.1)
SUN8	94.2	77.3	(78.3)	84.9	(85.8)	87.8	(90.2)
SUN9	96.2	78.6	(78.6)	88.8	(89.8)	87.9	(91.4)
SUN10	97.0	75.7	(81.3)	88.2	(90.0)	91.4	(93.3)
Mean	96.0	78.5	(81.8)	83.8	(87.1)	87.9	(90.8)

evidence variable is *IsA(Object$_i$,x)* with a value set to `true`. We then query each of the relations, *IsA(Object$_i$,s$_i$)*, *AtLocation(Object$_i$,s$_i$)*, *HasProperty(Object$_i$,s$_i$)*, and *UsedFor(Object$_i$, s$_i$)*. These queries return the probability of x's categorization, likely location, properties, and affordances; a query result is considered `true` if the associated probability is greater than 0.5, and `false` otherwise. Reverse inference uses *IsA* as a query variable (instead of as an evidence variable), enabling the robot to query for a list of objects that can be found at a particular location, have certain properties, or can be used for a desired purpose. Table 1 shows the results across all thirteen data sets, with reverse inference results listed in parenthesis.

Inference results for both the task descriptions and the SUN images showed high accuracies for the *IsA* relation, with an average of 96%. This is due to the fact that WordNet, the source of the *IsA* relation, is hand coded and thus contains highly accurate data. The accuracies for the *AtLocation*, *HasProperty*, and *UsedFor* relations average at 78.5%, 83.8% and 87.9%, respectively for forward inference, and 81.5%, 87.1% and 90.8% for reverse inference. All three query types show an increase in performance as the number of seed words increases, in part due to the improvements in the underlying disambiguation rate, and in part because of the richer set of connections present with more data.

Critically, this promising result shows that as few as 20 words obtained from a single image of the robot's environment already allows the robot to mine sufficient semantic information to predict many commonsense facts. Examples include identi-

crowd workers (0.8 agreement threshold). A comparison between hand-labeled and crowd-labeled data resulted in accuracy values within 1% for all tested instances.

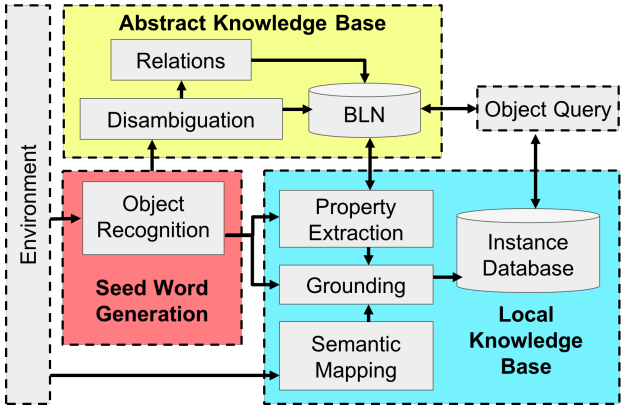


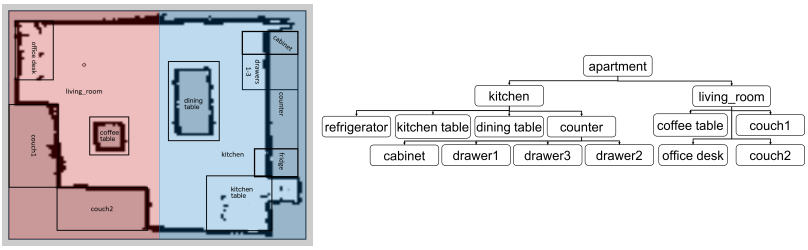
Fig. 5 Robot architecture overview.

fying a table, dishwasher, and cabinet as likely locations for a bowl; predicting that an apple has the properties red, round, and healthy; knowing that a knife can be used to pare, scratch, and separate; and inferring that a drawer can be used to contain, organize, and hide. In the next section we discuss how this general knowledge can be adapted to a specific robot environment.

5 Leveraging Semantic Knowledge in Situated Robotic System

To validate our Bayesian reasoning framework on a physical robot, we utilize the robot architecture shown in Figure 5. The architecture consists of three main components: Seed Word Generation, Abstract Knowledge Base, and Local Knowledge Base. Below, we present the technical details of each component, including improvements made to the BLN generation process to improve location data processing. In the next section, we report results from two trials with a mobile robot.

Object Recognition: The pipeline begins with object recognition, assigning object class labels (e.g., cup, bowl, etc.) to objects detected in the robot's environment. The generated class names become seed words that are used to extract information for the BLN. For object detection, we use the open source real-time object detection system YOLOv2 [14]. YOLOv2 uses a convolutional neural network and computes the location and classification of each object in an image in a single pass by dividing the image into cells, calculating an objectness score and then object classification probabilities over the individual cells, and then using anchor boxes to predict the object bounding boxes. For our robot experiments, we trained YOLOv2 on a subset of COCO [11] object classes, consisting of the following 31 objects commonly found in a home environment: *apple, banana, book, bottle, bowl, broccoli, cake, carrot, chair, clock, couch, cup, donut, fork, glass, knife, laptop, microwave, orange, oven, phone, pizza, plant, refrigerator, sandwich, sink, spoon, table, toaster, tv, vase*. Each time the system recognizes the object, the object label, bounding box of the object, and raw rectangle segment of the object is sent to the Local Knowledge Base, and the object labels are passed to the Abstract Knowledge Base.



Metric map Topological map

Fig. 6 Topological and metric maps of the robot's environment.

Abstract Knowledge Base (AKB): The Abstract Knowledge Base consists of the pipeline described in Section 3, with two modifications. First, we use the SUN image database as an additional source of *AtLocation* data, providing us with a richer set of room-level locations than available in ConceptNet. Locations of an object are imported based on probability of occurrence within the database. To maintain scope, we add only locations related to the high level environment the robot is operating in (i.e., related to *house*, eliminating *park*, *grocery store*, etc). Second, to ensure correct probability distribution of hierarchical location information (e.g., *counter-kitchen-apartment*), we encode transitivity properties using the first-order logic rules of the BLN.

Local Knowledge Base (LKB): We represent the robot's local environment through a collection of object instances, forming a memory of encountered items, and their locations and properties. For each object class o, we store i instances of that object within the LKB, where each instance corresponds to a unique object of that class. The LKB is implemented using PyTables and HDF5; each object class o is stored as a database, with a table generated for each object instance. We distinguish instances using multi-object tracking that identifies instances using local features. For each instance, we store the object label, previously seen locations (pose and semantic label), image region corresponding to the bounding box from object recognition, visual information (RGB-D values), and properties of the instance (e.g, color, material), if known. The resulting representation provides a scalable memory system that allows for efficient retrieval of all of its recent memories of instances.

To provide a semantic location for an object, we utilize a hybrid map [2], which links a topological map, consisting of a tree graph representing human domain knowledge, with a metric map of spatial locations in the environment. Figure 6 shows the topological and metric maps used in this work. The links between the topological map and metric map are expressed directly in the topological map nodes; association of each node with a volume in the metric map. This map structure enables the robot to obtain a semantic label for any 3D point that is hierarchical (e.g., object o is in a *drawer* in the *kitchen* in the *apartment*).

Fig. 7 Robot performing Case study 1. (a) scanning environment, (b) looking for bowl in refrigerator, (c) looking for and finding bowl in cabinet.

6 Evaluation in Robot Experiments

We evaluate our situated reasoning framework using two real-world case studies in a small-scale apartment setting (Figure 6) using a mobile robot equipped with a laser rangefinder and RGB-D camera.

Case study 1: In the first case study, we demonstrate the use of abstract semantic knowledge to enable the robot to locate objects never before seen in the environment. In this experiment, the robot begins by randomly exploring the environment shown in Figure 7, in which food items are visible on the table. The robot obtains the following 11 seed words from object recognition: *apple, banana, book, bottle, block, cup, donut, fork, orange, spoon, vase*, using which it constructs a BLN consisting of 97 nodes encoding the abstract knowledge of the environment. Next, we request the robot to find a *bowl* within the environment. Note that we provide the robot with the semantic map, shown in Figure 6, but no other information beyond the seed words listed, which do not include the *bowl*. To complete this task, the robot must first add the word *bowl* to the BLN, using the words already present in the network to perform word sense disambiguation. The result defines *bowl* as "a round vessel that is open at the top; used chiefly for holding food or liquids" as opposed to "a large structure for open-air sports or entertainments" or "a small round container that is open at the top for holding tobacco". Given this definition, appropriate relations are mined from WordNet and ConceptNet following previously described processes. Once the BLN has been updated, the robot performs a reverse inference query to locate the *bowl*, which results in the list of possible locations, ranked by probability value, shown in Table 2. Using the semantic map provided, the robot is able to ground the bolded words above to known locations in its environment. Using the resulting sorted list of locations (*refrigerator, cabinet, cupboard, sink, table*), the robot explores the environment until it successfully locates the *bowl* in the *cabinet*. This case study vali-

Table 2 Ranked list of likely *bowl* locations.

Location	Prob.
refrigerator	1.000
cabinet	1.000
cupboard	1.000
kitchen	0.933
dishwasher	0.786
closet	0.773
food	0.724
sink	0.723
kitchen_table	0.722
diningroom	0.273
bathroom	0.225

dates an important use case for the semantic reasoning framework in which situated and abstract knowledge is combined to enable the robot to reason about a previously unknown object.

Case study 2: In the second case study, we examine the relative tradeoffs of using abstract versus local knowledge in locating a known object given an increasing number of observations. We utilize a *bowl* and a *potted plant* as representatives of classes of objects that have a high and low location distribution, respectively. For this experiment, we allowed the robot to explore the environment and observe the location of the two items in 20 independent runs (Figure 8). During each run, the *plant* and *bowl* were placed in a random location sampled from the following distributions:

Fig. 8 The robot recording location of the bowl.

- *plant*: office desk: 50%, coffee table: 25%, counter: 25%
- *bowl*: dining table: 20%, kitchen table: 20%, counter: 12%, coffee table: 12%, office desk: 12%, drawer1: 12%, drawer2: 12%

Leveraging this data, we conducted 5000 simulated trials, splitting the dataset of the above observations into 75-25% randomized training-test set per trial. For each trial, we present our system with an increasing number of object observations and use the AKB and LKB to predict the location of the *bowl* and *plant*, comparing the result to the hold-out test set. We consider the trial to be a success if the true object location of the object is within the top 3 ranked locations returned by the algorithm.

Figure 9 presents the results of the experiment; solid lines represent inference using only local observations (instances stored in the LKB) and dashed lines represent inference performed using the Bayesian Logic Network based on abstract knowledge. The two knowledge bases share no information to illustrate the relative strengths and weaknesses of each. For both objects, the BLN provides the robot with the ability to make an "intelligent guess" when no other information is available, allowing the robot to predict the correct location of both *plant* and *bowl* with 60% accuracy. The robot's local observations, by comparison, become highly valuable as the robot gains information about the environment, surpassing BLN performance after 2 and 11 observations for the *plant* and *bowl*, respectively.

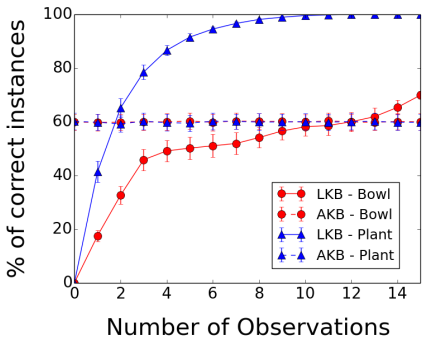


Fig. 9 Average accuracy (AKB vs. LKB) across 5000 permutations to predict the locations of *potted plant* and *bowl*.

As this experiment demonstrates, the incorporation of semantic knowledge into the robot architecture provides a way to supplement or guide local observations. Just

as a person entering an unfamiliar house can leverage prior semantic knowledge to guess likely item locations (e.g., search kitchen drawers for a fork), our robot is able to infer likely object locations. Over time, a person would stop relying on abstract information and instead utilize local knowledge, going directly to the needed drawer without searching. This process is equivalent to enabling the robot to increasingly rely on the LKB. Our future work will explore techniques for merging abstract and local data sources, as well as ways in which the robot is able to leverage abstract knowledge to guide exploration, semantic mapping, and task execution in unfamiliar environments.

7 Human Refinement using Actively Situated Knowledge (ASK)

The final component of our work, is to consider refinement of the robot's abstract knowledge in the BLN from another situated knowledge source – a co-present human. Our prior case studies demonstrated the effectiveness of the BLN in predicting location information. However, certain other types of object information, such as material properties and affordances, are more difficult for the robot to acquire independently. Additionally, the BLN contains a certain level of noise, due to both word sense disambiguation errors and noise within ConceptNet. We, therefore, introduce Actively Situating Knowledge (ASK), a technique which situates the general knowledge in the BLN with the help of a human user[2], which we apply to the *HasProperty* and *UsedFor* relations in the BLN.

Note that the BLN contains too many property edges to make it practical to verify each one with the human user. Thus, our goal is to intelligently query a subset of property relations for validation while still obtaining verification of all the relations within the BLN. For this, we first modify the BLN to add inter-property edges. For all pairs of properties in the BLN, we add an edge if a relation exists between them in ConceptNet. For example, since an *IsA* relation exists between Metal and Aluminum in ConceptNet, and a *HasProperty* relation exists between Glass and Recyclable, inter-property edges are added between these pairs of relations, respectively.

Next, given N, the number of known objects, and P, the number of known properties, we define:

- $T_{material}$: a $Px1$ table listing all material properties present in our BLN (i.e., nodes that hold a relation with *material* in the ConceptNet, e.g. *metal, aluminum, plastic*, etc.).
- T^{O}_{assoc}: a NxP table storing the ConceptNet association values between object O and every property belonging to O, where the the association value is a measure between 0 to 1 of how related two words are[3].
- $T^{O}_{interprop}$: a PxP table storing the ConceptNet inter-property association values between any two properties, P_O, associated with object O.

[2] This approach could also be combined with crowdsourcing, although we relied on a co-present expert for all experiments described here.

[3] We ignore properties with value < 0.07

Given the above tables, we select properties to verify with a human expert based on the inter-property association value. For each object, we alternate between querying the expert about property $p \in P_O$ with the highest association value and the lowest association value in T^O_{assoc}. We alternate queries between maximum and minimum inter-property association value in order to maximize information gain by asking about properties that are least likely to occur together.

If a property is confirmed as `true` by the user, and exists in $T_{material}$, then all other material properties belonging to that object are assumed to be `false` and are not queried (we make the simplifying assumption that each object only has a single material property). Given this information, we can also infer that the predecessors of that property are true for O (e.g., if *aluminum* is true, then *wood* is false but *metal* is true). For the successors of O, we assume their *hasProperty* relations are true (e.g., if *metal* true, then *opaque* true), but that *IsA* relations must still be verified (e.g., if *metal* true, still need to verify *aluminum*). If a node in this *isA* set is verified to be `true`, the rest are assumed to be `false`. However, if an attribute is verified to be *false*, all its *IsA* successors are assumed to be *false* but its *HasProperty* successors are still queried (e.g., if *metal* is `false`, then *tin* is assumed to be `false` but *recyclable* is still queried). We repeat this process until all the properties are verified as `true/false` and create an updated, expert-verified BLN (*vBLN*) using the updated true verified property relations.

We evaluate ASK by comparing the accuracy of the resulting vBLN to a ground truth BLN, gBLN, generated by an expert. The gBLN differs from vBLN in that every edge was expert-verified instead of just the subset explored by ASK. To evaluate ASK performance we calculate the dissimilarity index:

$$I_{dis} = \frac{\text{Uncommon edges between ground truth, gBLN and vBLN}}{\text{Total number of unique edges in gBLN and vBLN}}$$

To evaluate ASK, we trained a BLN using the 31 seed words from the COCO dataset, resulting in a network consisting of 195 property edges. Using ASK, we performed 84 clarifications, resulting in 50 pruned property edges in the vBLN. While this is a large number of clarifications, during a deployment such queries could occur over a length of time (multiple days) as the robot spends time learning about its environment. The algorithm obtained a final dissimilarity score $I_{dis} = 0.11$, indicating that the vBLN contained only 6 extraneous edges in comparison to the ground truth. In future work, this result can be further improved through selective filtering of data imported from ConceptNet, or through the use of more reliable data sources. The above experiment demonstrates that even with highly noisy data sources, we are able to intelligently refine the robot's semantic knowledge. Furthermore, the inference techniques used in ASK can be leveraged in combination with the robot's own sensing in addition to human input, such as if material information were to become available.

8 Conclusion

In this paper, we explore the ability of a robot to use limited local information, sometimes as little as the set of object labels recognized in a single image, to automatically mine semantic information about its environment. We embed the mined information within a statistical relational model, and demonstrate its use on both abstract tasks and as part of a robot architecture, as well as introduce a technique for interactively refining the semantic knowledge with the help of a human expert. Our results show that, just as for humans, semantic knowledge can provide valuable guidance in the absence of extensive familiarity with the operating environment. In future, we will explore how local and abstract information can be merged over time to enable the knowledge framework to further adapt to the robot's current environment.

9 Acknowledgement

This work is supported in part by NSF IIS 1564080 and ONR N000141612835.

References

[1] Adrian Boteanu and Sonia Chernova. "Solving and Explaining Analogy Questions Using Semantic Networks". In: *AAAI Conference on Artificial Intelligence*. AAAI. 2015, pp. 1–8.

[2] Pär Buschka and Alessandro Saffiotti. "Some notes on the use of hybrid maps for mobile robots". In: *Proc. of the 8th Int. Conf. on Intelligent Autonomous Systems*. 2004, pp. 547–556.

[3] Junpeng Chen and Juan Liu. "Combining ConceptNet and WordNet for Word Sense Disambiguation". In: *International Joint Conference on Natural Language Processing*. 2011, pp. 686–694.

[4] Jia Deng et al. "Imagenet: A large-scale hierarchical image database". In: *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE. 2009, pp. 248–255.

[5] Robert Fung and Kuo-Chu Chang. "Weighing and Integrating Evidence for Stochastic Simulation in Bayesian Networks". In: (Mar. 2013).

[6] Evgeniy Gabrilovich and Shaul Markovitch. "Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis". In: *International Joint Conference on Artificial Intelligence*. 2007, pp. 1606–1611.

[7] Dominik Jain, Bernhard Kirchlechner, and Michael Beetz. "Extending markov logic to model probability distributions in relational domains". In: *KI 2007: Advances in Artificial Intelligence*. Springer, 2007, pp. 129–143.

[8] Dominik Jain, Stefan Waldherr, and Michael Beetz. *Bayesian Logic Networks*. Tech. rep. Technische Universität München, 2009, pp. 1–21.

[9] H. Kucera and W.N. Francis. *Computational Analysis of Present-Day American English*. Providence: Brown University Press, 1967.

[10] Douglas B Lenat. “CYC: A large-scale investment in knowledge infrastructure”. In: *Communications of the ACM* 38.11 (1995), pp. 33–38.

[11] Tsung-Yi Lin et al. “Microsoft COCO: Common Objects in Context”. In: *CoRR* abs/1405.0312 (2014).

[12] Cynthia Matuszek et al. “An Introduction to the Syntax and Content of Cyc.” In: *AAAI Spring Symposium: Formalizing and Compiling Background Knowledge and Its Applications to Knowledge Representation and Question Answering*. Citeseer. 2006, pp. 44–49.

[13] George A. Miller. “WordNet: A Lexical Database for English”. In: *Communications of the ACM* 38.11 (1995), pp. 39–41.

[14] Joseph Redmon et al. “You only look once: Unified, real-time object detection”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 779–788.

[15] Matthew Richardson and Pedro Domingos. “Markov logic networks”. In: *Machine Learning* 62.1-2 (Feb. 2006), pp. 107–136.

[16] Ashutosh Saxena et al. “Robobrain: Large-scale knowledge engine for robots”. In: *arXiv preprint arXiv:1412.0691* (2014).

[17] Nuno Seco et al. “An Intrinsic Information Content Metric for Semantic Similarity in WordNet”. In: (2004).

[18] Robert Speer and Catherine Havasi. “Representing General Relational Knowledge in ConceptNet 5”. In: *Proceedings of the Eight International Conference on Language Resources and Evaluation*. Istanbul, 2012.

[19] Emilia Stoica and Marti A. Hearst. “Nearly-Automated Metadata Hierarchy Creation”. In: *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2004, pp. 117–120.

[20] Moritz Tenorth and Michael Beetz. “KnowRob–knowledge processing for autonomous personal robots”. In: *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE. 2009, pp. 4261–4266.

[21] George Tsatsaronis, Iraklis Varlamis, and Michalis Vazirgiannis. “Word Sense Disambiguation with Semantic Networks”. In: *Text, Speech, and Dialogue*. Ed. by Petr Sojka et al. Springer, 2008, pp. 219–226.

[22] Markus Waibel et al. “Roboearth”. In: *IEEE Robotics & Automation Magazine* 18.2 (2011), pp. 69–82.

[23] Jianxiong Xiao et al. “SUN database: Large-scale scene recognition from abbey to zoo”. In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, June 2010, pp. 3485–3492.

[24] Yuke Zhu, Alireza Fathi, and Li Fei-Fei. “Reasoning about object affordances in a knowledge base representation”. In: *European conference on computer vision*. Springer. 2014, pp. 408–424.