International Journal of Social Robotics manuscript No.
(will be inserted by the editor)

Multimodal object-based environment representation for assistive robotics

Yohan Breux · Sebastien Druon · Jean Triboulet

Received: date / Accepted: date

Abstract Autonomous robots are nowadays successfully used in industrial environments, where tasks follow predetermined plans and the world is a known (and closed) set of objects. The context of social robotics brings new challenges to the robot. First of all, the world is no longer closed. New objects can be introduced at any time, and it is now impossible to build an exaustive list of them nor having a precomputed set of descriptors. Moreover, natural interactions with a human being don't follow any precomputed graph of sequences or grammar. To deal with the complexity of such an open world, a robot can no longer solely rely on its sensors data : a compact representation to comprehend its surrounding is needed.

Our approach focuses on task independent environment representation where human-robot interactions are involved. We propose a global architecture bridging the gap between perception and semantic modalities through instances (physical realizations of semantic concepts).

In this article, we describe a method for automatic generation of object-related ontology. Based on it, a practical formalization of the ill-defined notion of "context" is discussed. We then tackle human-robot interactions in our system through the description of user request processing. Finally, we illustrate the flow of our model on two showcases which demonstrate the validity of the approach.

Keywords knowledge representation · human-robot interaction · natural language processing · assistive robotics

Yohan Breux · Sebastien Druon · Jean Triboulet
Laboratory of Informatics, Robotics and MicroElectronics, University of Montpellier, 161 rue Ada, 34095 Montpellier, France
E-mail: {breux,druon,triboulet}@lirmm.fr

1 Introduction

In the previous decades, efforts have been made to understand and exploit the social benefits of robots in human environment [24]. In particular, some applications are focused on therapeutics [41, 54, 30, 17], education [19, 3] and human-robot cooperation [43, 4]. Unlike industrial applications where the environment is controlled and the interaction with the human operator limited, such applications require the robots to have a deeper understanding of their surroundings. Furthermore, these interactions are made easier for the operator when performed through oral interactions.

Because of an early development towards industrial applications, majority of researches in robotics are task oriented and focus their efforts on action descriptions [51, 52]. They use specific predefined environment representations for the tasks at hand. However, these representations lack genericity and can't compactly represent abstract knowledge about the robot surroundings. Furthermore, as underlined above, robots and human should share a common way to describe the world and its concepts through natural language.

It is important here to define the meaning of "environment representation" as it depends on the application context. We consider three main categories of representation :

- **Geometrical** representation where the environment is modelized as a set of primitives (point, surface, volume) eg. occupancy grid in SLAM [53, 55].
- While geometrical representation suits applications such as exploration or obstacle avoidance, it can't be used for objects manipulation as it models the world as a unique "block". In this case, clustering of primitives into objects hypothesis is required. This leads to an **object-based** representation.

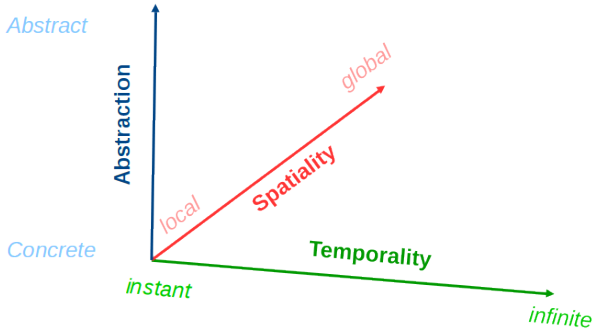


Fig. 1: Axis characterizing environment representation

– Human beings, through natural language, have defined a **semantical** representation of their surrounding. Unlike the previous representation based on perception, it can compactly describe unseen objects. It also allows a robot to reason about facts it has not experienced yet. Besides, this representation is fundamental for human-robot interaction.

We propose to characterize each representation according to the three axis shown in Figure 1 :

– **Spatiality** (local-global) : an image of a room would typically be a local representation whereas a map of a building would be considered as global. In semantics, some facts can be true only locally eg. *taxi are yellow* which is true in New York but not in general.
– **Temporality** (instant-infinite) : any model has a validity period. Thus an image can only be considered as a valid representation the instant it was taken. The existence of some instances eg. *my glasses* is valid on a limited period of time whereas general facts such as *glasses are used for vision correction* are timelessly true.
– **Abstraction** (concrete-abstract) : concrete raw sensors data and abstract semantic concepts.

Based on the observation above, we propose a generic and task independent architecture for environment representation. The rest of the paper is organized as follows. First, section 2 provides a quick overview on our multimodal architecture. In section 3 we discuss on the state of the art for environment representation. The section 4 is the main focus of this paper. It describes our method for automatic ontology generation which is the basis of the semantic modality in our architecture. We define the important notion of context based on this ontology in section 5. Section 6 deals with human-robot interactions through the generation of requests from natural language. Finally, section 7 illustrates the global flow of our model on two showcases to demonstrate the validity of our approach.

2 Environment modeling : a multimodal approach

In this work, we propose a three-layer model covering all the representation categories described previously (Figure 2). It is composed of a perception, instance and knowledge unit. The perception module processes low-level raw sensors data. It is responsible for generic scene segmentation, instance localization (tracking) and concepts detection whereas the knowledge module represents high-level semantic relations expressed in natural language. The instance unit bridges the gap between those possibly contradictory modalities by linking real-world observations to generic semantic concepts. Furthermore, instances are related to each other through the tasks in which they are involved (Figure 3).

This paper is an extension of our previous work [8]. We propose an heuristics-driven method for automatic generation of an object-based ontology from dictionary definitions. This is motivated by the fact that, at the best of our knowledge, there is no generic yet expressive ontology for objects description. Existing ontologies are either

– Limited for specific applications. It results in shallow ontologies difficult to reuse.
– Too broad for robotics by covering domains such as History or Politics. This is the case of OpenCyc [25, 27] or DBpedia [2].

In both cases, they are lacking details on concepts related to physical objects as it is shown in section 4.

Based on the constructed ontology, we then attempt to give a formal and computable definition of what is commonly called as *"context"*. It can be informally defined as a "confounding" set of concepts explaining "why" some observational correlations occur. An illustrated example is given in Figure 9. The idea here is to exploit *causality* information provided by semantics.

Semantic representation is also directly involved in human-robot interactions. We discuss how natural language inputs are processed and used to respond to simple user requests. In particular, we present three use cases illustrating the general flow of our system.

3 Related work

The first need for an environment model is usually for the robot to locate itself. This is why, in the past decade, Simultaneous Localization and Mapping (SLAM) [53, 55] was the prominent method for environment modeling in autonomous robotics. Depending on available sensors, it provides 2D/3D geometric representation of environment based on occupancy grid or unstructured

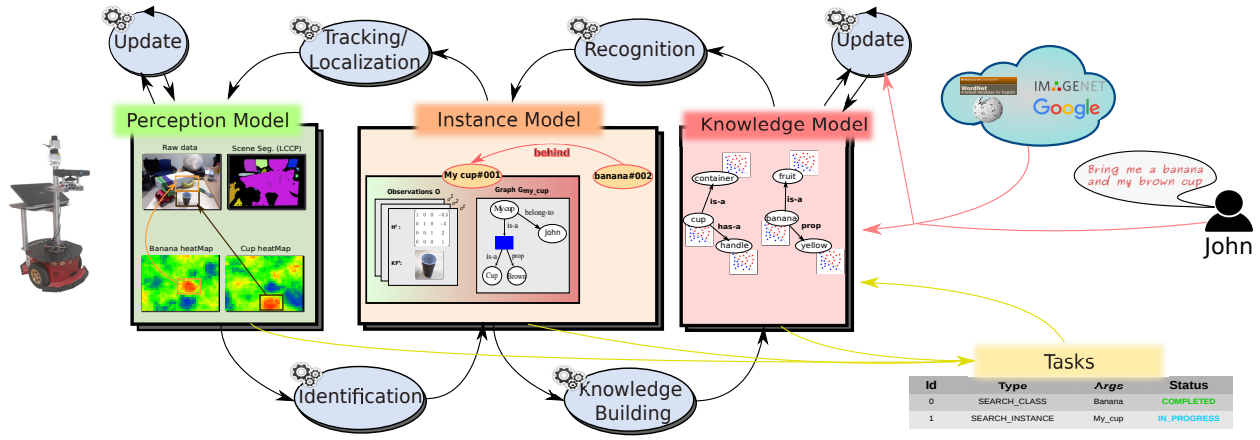


Fig. 2: Illustration of our three layer environment representation built from a user request

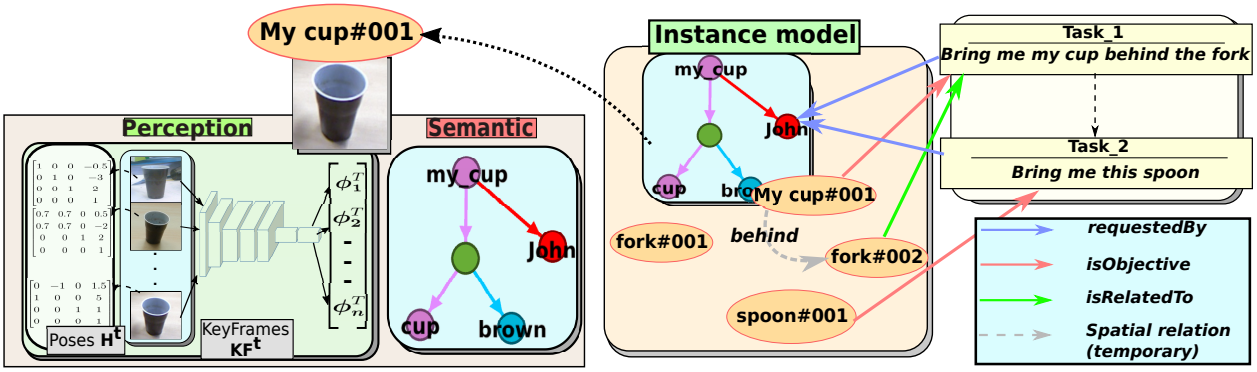


Fig. 3: Illustration of the Instance model (best viewed in color)

point cloud. More recently, Semantic mapping aims at enriching the environment representation by clustering primitives into meaningful objects using recent advances in deep machine learning.

Sunderhauf et al. [48] propose a mobile robot system for categorizing places based on Convolutional Neural Network (CNN) [20]. They use a modified version of occupancy grid where a vector of class probabilities is attached to each cell. They leverage this information to improve objects detection using prior probabilities on the place category. The approach in [49] of the same authors consists in merging geometrical and object-based representation of the environment. They continuously (partially) structure the raw point cloud map obtained with ORB-SLAM2 [31] using CNN for object detection and classification. Each detection is then used to build or update the 3D model of corresponding object.

Those representations are solely focused on direct observations. While those approaches are practically interesting, they don't leverage semantic information provided by their labeling. In our work, we are particularly interested in multilayer methods using jointly semantic and geometric information.

Pronobis et al. [37, 38] introduce a multilayer architecture for indoor semantic mapping by reasoning on heterogeneous data such as object occurrences or room dimensions. The first low-level layer is an occupancy grid which is spatially discretized into places in the second layer. Each place is defined on object occurrences and geometrical aspects. The last conceptual layer is an ontology linking semantic concepts and instances detected in the environment eg. "A living-room **has-a** TV" or "place1 **has-a** instance1". This ontology is then casted as a probabilistic graph for inference.

Lang et al. [22, 21, 23] propose a similar architecture for outdoor semantic mapping. Their first layer is a point cloud map which is first over-segmented into cluxels (cluster of voxels) further grouped into Observations. Each Observation is classified into classes (horizontal plan, scatter appearance). Finally, cluxels are grouped into objects based on observation labels and an ontology with relations such as *tree has-a scatter appearance*.

Those two frameworks leverage both visual and semantic modalities but are mainly centered on the mapping process. They both lack deeper understanding on

relations between objects of the scene and are difficult to scale up on an open world.

Some approaches are grounded on knowledge presentation. KnowRob [51, 52] is a task-oriented system leveraging semantic knowledge in the context of human assisting robot. They use a knowledge base bootstrapped on OpenCyc ontology [25, 27]. Visual inferences are made at run-time by using *computables* which are called when the attached concept is part of a query. The system is essentially focused on action descriptions. Object descriptions are limited to the minimum required for their experiments (Figures 4a, 4b).

A large part of the literature attempts to increase algorithms performance on a close-world assumption through a variety of datasets. However, robots evolving in a partially unknown human environment have to reason with generic and adaptable models. The guiding idea behind our work is to complement, when possible, *correlations* inferred from observational data with *causalities* extracted from semantic knowledge as illustrated in Figure 9. Occurrence of *fork* in *kitchen* is a correlation expressed by the conditional probability $p(c = fork|r = kitchen)$. In fact, this occurrence can be explained by their common context : *eating food*. The objective is to leverage such facts to transfer the observational correlation to unseen concepts sharing the same context eg. occurrence of fork in restaurants.

In the following section, we first present our approach to build an object-based ontology and then use it to extract context of concepts.

4 An automatically generated object-based ontology

Our knowledge model is an ontological graph composed of object (*fruit*, *cup*) and action (*eat*, *serve*) concepts. In addition to classical relations such as *isA* (hyper-/hyponymy) and *hasA* (meronymy), it provides relations expressing property and usage. It is a compact representation of the environment as defined by human through natural language. There is already a large variety of ontologies in the literature. For instance, OpenCyc is an example of global ontology built manually by experts whereas DBpedia [2] extracts structured information from Wikipedia. For environment representation purpose, those ontologies are too broad by describing domains such as History or Politics. Meanwhile, they do not provide enough details for object descriptions. NELL (Never Ending Language Learning) [10, 29] is a project aiming at continuously parsing Internet to automatically build an ontology. Although the results are promising, it is not adequate yet for robotic applications. Figure 4 illustrates some problems with existing ontologies. The subgraphs corresponding to the *fork* and *cup* concepts were extracted from NELL and KnowRob. We observe some overspecialized concepts such as *sippy cup with removable lid and two handles* or *expresso cup with saucer*. Some relations such as

$$< water\ glass\ cup, generalization, bathroomitem >$$

can hold but are not relevant in general. Besides, some simple relations are not present eg.

$$< salad\ fork, generalization, fork >$$

Although there is no error-free methods, we can see that there are only few information usable by a robot and it is mainly connected to spatial context (eg *kitchenitem*, *bathroomitem*). In comparison, the subgraphs 4e and 4f extracted from our ontology show a variety of relations related to object usage with the emergence of what we call *context* concepts such as *food* in the fork subgraph. At the best of our knowledge, there is no ontology focused on object descriptions adapted for robotic applications. This motivates our development of an automatic ontology generation method explained in the following section.

4.1 An ontology based on dictionary definitions

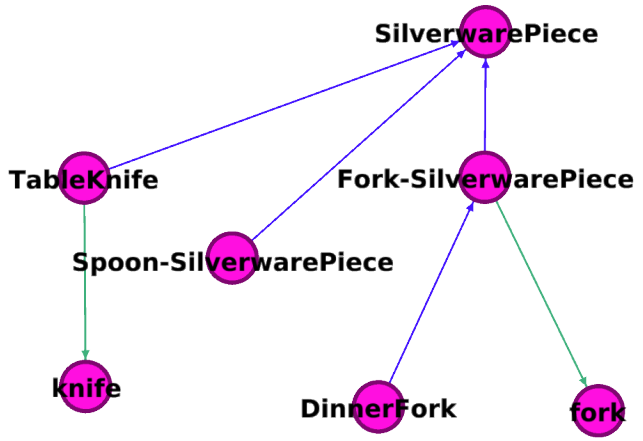
We generally think about physical objects in terms of physical appearance (*"a bottle has a cylindric shape"*) and/or usage (*"a bottle is used to hold liquid"*). We also want a minimum of base concepts in our ontology to avoid the overspecialization as seen in Figure 4. As much as possible, complex concepts must be represented as a combination of basic ones. Thus dictionary definitions are a natural source for creating such an ontology. For instance, Cambridge Dictionary [1] gives for *fork* :

- First sense : *a small object with three or four points and a handle, that you use to pick up food and eat with.*
- Second sense : *a tool with a long handle and three or four points, used for digging and breaking soil into pieces.*

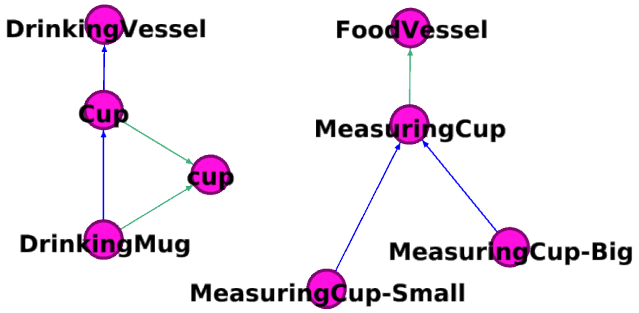
We clearly see that, for both senses, fork is defined by its physical appearance (*small*, *with a long handle*) and its usage (*use to pick up food*, *used for digging*). In our work, we choose to bootstrap our ontology on the lexical database WordNet [28] and extends it by syntactically analyzing its definitions.

We are not the first trying to exploit WordNet definitions. Novischi [33] aims at disambiguating words in

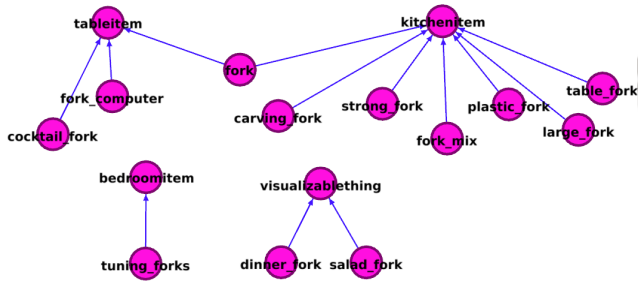
[1] https ://dictionary.cambridge.org/dictionary/english/fork



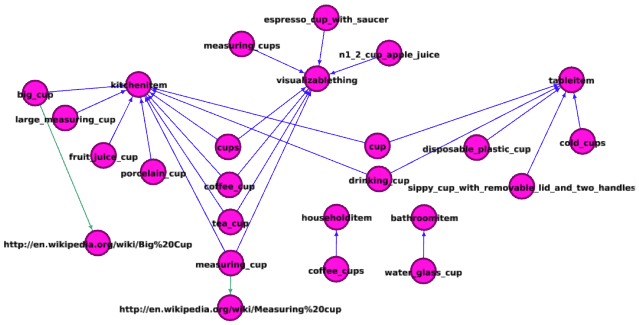
(a) fork-KnowRob



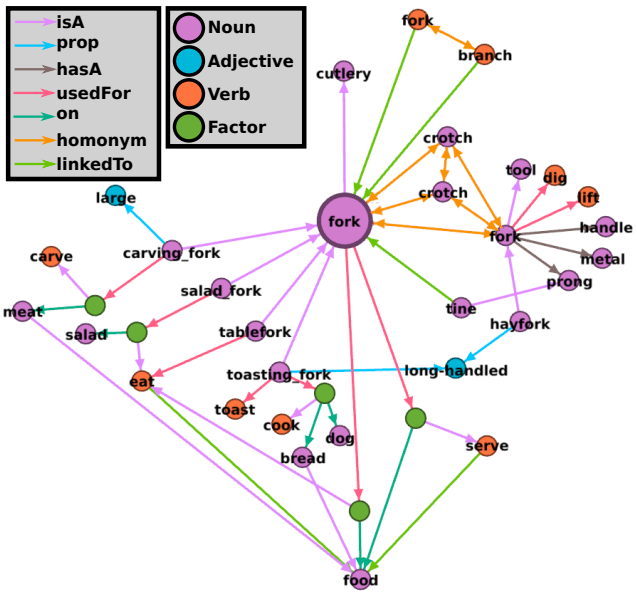
(b) cup-KnowRob



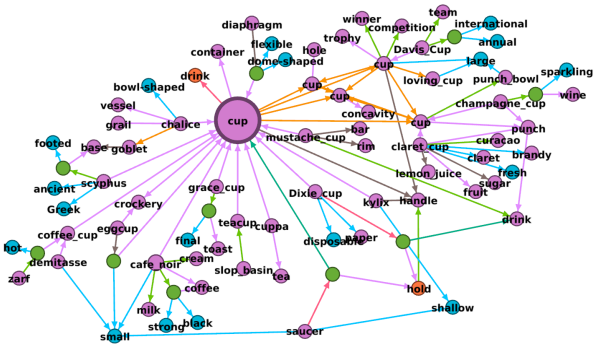
(c) fork-NELL



(d) cup-NELL



(e) fork-ours



(f) cup-ours

Fig. 4: Comparison of KnowRob, NELL and our ontology for *fork* and *cup* concepts. Relations for KnowRob : *subClassOf* (blue) and *hasValue* (green). Relations for NELL : *generalizations* (blue) and *haswikipediaurl* (green). For our ontology, the subgraphs are the set of nodes at distance 1 (excluding factor node) from the concept node and its hyponyms (best viewed in color)

Table 1: WordNet categories for each Part-Of-Speech (POS). We underline categories used in this work

POS	Category
Noun	act, animal, artifact, attribute, body, cognition, communication, event, feeling, food, group, location, motive, object, person, phenomenon, plant, possession, process, quantity, relation, shape, state, substance, time
Adjective	all,pert (pertainym), ppl (participial)
Adverb	all
Verb	body, change, cognition, communication, competition, consumption, contact, creation, emotion, motion, perception, possession, social, stative, weather

WordNet definitions by prioritizing accuracy at the expense of coverage. He builds patterns by considering successive words in the definition set, which are further merged or filtered according to their occurrences and some heuristics. The best corresponding pattern is searched for each word of every definition. The word is then assigned the same sens as the one it has in the previously found pattern. The proposed approach yields a 98% accuracy for a limited coverage of 6.6%. This shows in particular that disambiguation is a rather difficult problem. Currently we bypass it by keeping homonym relations in our graph, allowing us to disambiguate on the fly depending on the current context. DeBoni et al. [12] extracts telic relations (equivalent to our *usedFor*). They consider a set of patterns corresponding to the searched relations. Let O be a word of a relation extracted from the definition $D_{O'}$ of a word O'. D' is defined as the set of words coming from D_O and D_{O_h} where O_h are hyper/hyponyms of O. O is then disambiguated using a similarity measure based on co-occurrences of words in $D_{O'}$ and D'. They obtain 60% accuracy on a small WordNet sample (10%). Bracewell and al. [5] also describe a pattern matching approach to extract knowledge on causal agent in WordNet. However, they do not consider disambiguation problem in their work.

4.2 WordNet

WordNet [28] is a manually generated lexical English database. Words corresponding to the same concept (synonyms) are grouped in a structure called *synset* with a unique identifier (WordNetId) and a definition. Synsets are also grouped depending on their Part-Of-Speech (POS) and on their category as enumerated in Table 1. The WordNetId starts with a letter (n for noun, a for adjective and v for verb) followed by a sequence of 8 digits. In this paper, concepts are written in the form *concept-WordNetId* to disambiguate homonyms such as *fork-n03383948* (hypernym of *tablefork*) and *fork-n03384167* (hypernym of *hayfork*).

WordNet ontology is a knowledge graph where nodes correspond to synsets and edges are semantic relations, mainly hyper/hyponymy (is-a) and meronymy (has-a). However, some relations appearing in definitions are not represented in the ontological graph. For instance, *cup-n03147509* is defined as *small open container usually used for drinking; usually has a handle* but the *handle* concept is not considered as a meronym (part) and there is no relation with the *drinking* action. We extract those relations with the method presented in section 4.4.

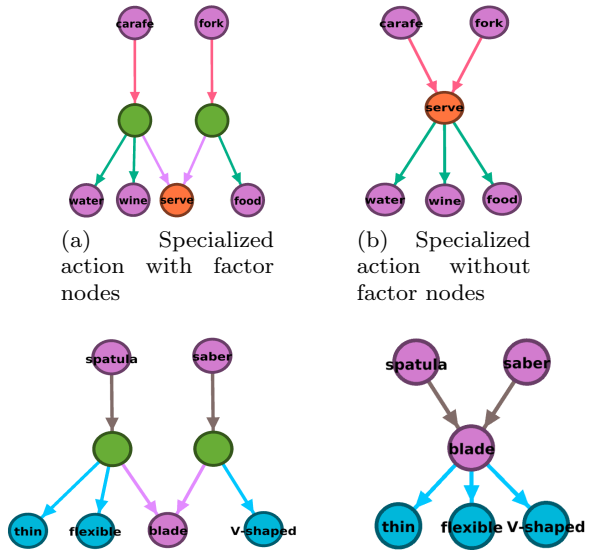
4.3 Definition of the ontology graph

We define our ontology graph by a mixed factor graph $G_K = \{V_K, E_K\}$. $V_K = V_K^C \cup V_K^F$ represents the set of nodes, which can be a concept (V_K^C) or a factorization (V_K^F) of concepts. There are two types of concepts : Object-Property V_K^O or Action V_K^A. Thus we have four kinds of vertex : concept of physical object or property $V_K^{C,O}$, concept of actions $V_K^{C,A}$, and those used to factorized object $V_K^{F,O}$ and action concepts $V_K^{F,A}$. Edges E_K can be of the following types :

- **isA** : expresses hyper/hyponymy eg.
  isA(cup, container), *isA(paper_cup, cup)*.
- **hasA** : represents part of objects (meronymy) eg.
  hasA(cup, handle).
- **prop** : represents properties of concept, usually expressed by an adjective but can be noun in case of material eg.
  prop(wheel, circular), *prop(desk, wood)*.
- **usedFor** : represents the function of a concept which is expressed by a verb of action eg.
  usedFor(cup, drink).
- **on** : if such information is available, used to precise the object on which the action applied eg.
  on(drink, tea).
- **linked-to** : represents other unknown relations eg.
  linkedTo(hold, hand).
- **homonym** : obtained by linking synsets with at least one common word. They are substantial (see Table 3a).

All edges are directed except for the homonym relation.

Unlike previous ontologies, we use factor nodes in our graph. Their advantages is two-fold. First, it allows to represent combination of concepts without explicitly creating (and naming) a new concept. More impor-



(a) Specialized action with factor nodes

(b) Specialized action without factor nodes

(c) Specialized object with factor nodes

(d) Specialized object without factor nodes

Fig. 5: Illustration of the importance of factor nodes in our ontology

tantly, they are essential for keeping information consistent in our graph. To demonstrate this, we consider the following pairs of concepts : (*fork-n03383948*, *carafe-n02960903*) and (*spatula-n04270147*,*saber-n04121511*). In Figure 5a, we can see the fact *fork is used to serve food* which is expressed by

$$usedFor(fork, fn) \wedge isA(n, serve) \wedge on(fn, food)$$

where $fn \in V_K^{F,A}$ represents an *action specialization*. Suppose now that we remove this factor node and link directly *fork* to *serve* : we also have to link directly *food* to *serve* so that we obtain

$$on(serve, food) \wedge usedFor(fork, serve)$$

as illustrated in Figure 5b. It is clear that if *serve* is linked to other target objects (eg. *wine* for the *carafe* concept), we can't determine on which objects *fork* can be used. Similarly, Figure 5c shows examples of *object specialization* factor nodes ($V_K^{F,O}$) and Figure 5d illustrates how the representation would be like without factor nodes.

4.4 Generation of ontology from WordNet definitions

In this section, we discuss how we build our ontology based on WordNet definitions. Figure 6 outlines the different steps. First, we initialize our ontological graph on WordNet ontology based on hyper/hyponymy (isA) relations. As some categories of words are not relevant

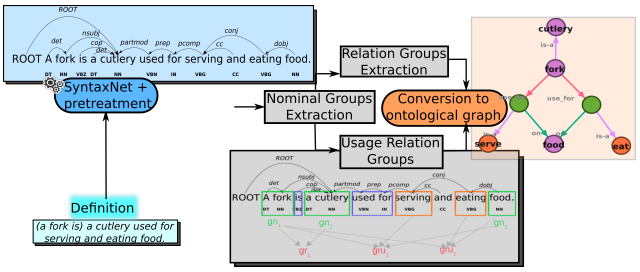


Fig. 6: Flow of the proposed approach for ontology construction

for the robotic field, we limit our work on a subset underlined in the Table 1. Then we extract the definitions attached to each synset. They are syntactically analyzed using a pre-trained transition-based neural network SyntaxNet [1]. Our approach is bottom-up : we cluster words into Nominal Groups (NG) which are further grouped as Relation Groups (RG) or Usage Relation Groups (URG). Finally, we convert this segmentation into a graph representation. In practice, our ontological graph is stored as a list of Prolog facts.

4.4.1 Syntactical analysis

The first step of our method consists in converting definitions (set of words) in a richer representation. We use the network SyntaxNet which returns data in the CoNLL-X format [9] as illustrated in Figure 7. In brief, it assigns a Universal (coarse)/ language-specific (fine) Part-Of-Speech (resp. UPOS [36]/XPOS [42]) to each word of the input sentence. Universal Dependency Relations (UDR) [13, 14, 32] express grammatical relations between words such as *dobj* (direct object). Once this representation obtained, we post-process it by adding the lemma of each word using WordNet's morphological processing. We remove some useless patterns such as *kind/piece/type/variant of*. Comma related to conjonction are replaced by the same conjunction (*and* or *or*). Compound word POS is not used by SyntaxNet so we use heuristics when two successive nouns have a dependency relation. We then check if the potential compound word exists in the WordNet database. Finally, another heuristics are used for correcting -ing verbs wrongly labeled as noun.

4.4.2 Nominal Groups and (Usage) Relation Groups Decomposition

We segment the input sentence S in the CoNLL-X format into a set of nominal groups NG (noun, determinant and set of related adjectives). Constraints between NGs expressed by conjunction such as *and* and

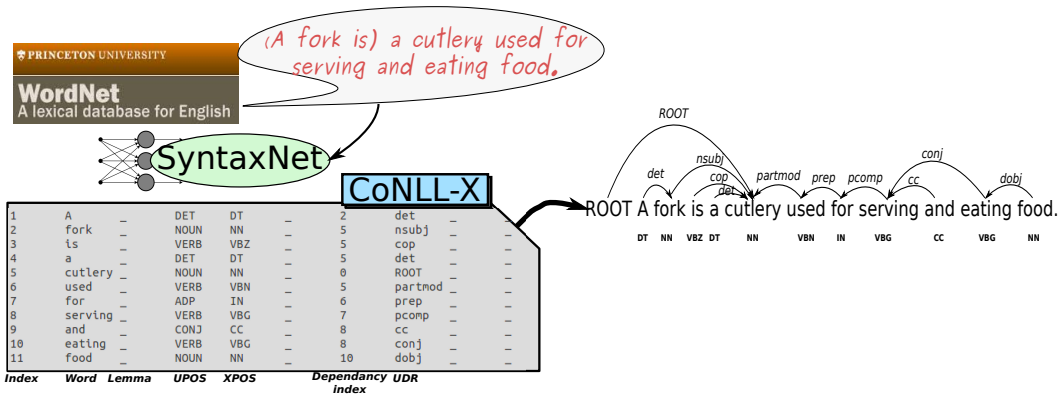


Fig. 7: Example of SyntaxNet output on WordNet definition for *fork-n03383948*

Table 2: Patterns employed to convert URG and RG into relations

Relation	Predicate	XPOS	Preposition	COD UPOS
prop	make	VBN, VBZ, VBP	of	NOUN
hasA	have	VBP, VBZ, VBG	of	NOUN
	consist	VBP, VBZ, VBG	of	NOUN
	design	VBN	with	NOUN
	equip	VBN	with	NOUN
	support	VBN	on,by	NOUN
	with	IN		NOUN
useFor	design	VBN	for	VERB
	use	VBN	for,to	VERB

or are also extracted. We also propagate conjunction repeated by a comma such as in *"a banana is a yellow, green or red fruit."*. Those constraints are set of set of NGs noted $C^{\wedge}_{NG}$ (resp. $C^{\vee}_{NG}$). We then obtain in the *banana* example NG_0 = "a banana", NG_1 = "yellow fruit", NG_2 = "green fruit", NG_3 = "red fruit" and $C^{\vee}_{NG} = \{NG_1, NG_2, NG_3\}$, $C^{\wedge}_{NG} = \emptyset$.

NGs are then further grouped into Relation Groups (RG) which are triplets of the form

$$< NG_{subject},\ Predicate,\ NG_{object} >$$

Constraints $C^{\wedge}_{NG}$, $C^{\vee}_{NG}$ are also propagated to RG. We then have

$$RG_i = < NG_0, be, NG_i >,\ i \in \{1,2,3\} \quad (1)$$
$$C^{\vee}_{RG} = \{RG_1, RG_2, RG_3\},\ C^{\wedge}_{RG} = \emptyset$$

We also define Usage Relation Group (URG) for representing *usedFor* and *on* relations. Indeed, for predicate corresponding to usage (eg. *designed for*, *used to*), relations are of the form

$$< NG_{subject},\ Predicate,\ Verb,\ Targets >$$

where $Targets$ are direct objects of $Verb$.

4.4.3 Conversion to an ontological graph

Finally, we have to convert the segmented sentence into an ontological graph. A node is created for each analyzed synset and for each word (noun, adjective and verb) appearing in one of the structure (NG, RG and URG). When a NG has at least one adjective, it is represented as a factor node. This is the case for the NG *a heavy rigid head* in Figure 8. Similarly, a factor node is created to represent URG with at least one direct object. Homonym edges are created between synsets with one homonym in common. Table 2 shows the patterns employed for the different relations. Finally, Figure 8 is a full illustration of the process for the concept *hammer-n03481172*.

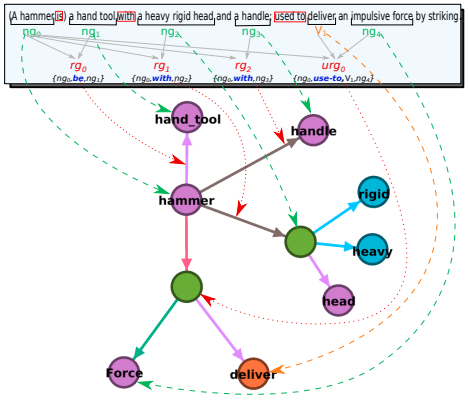


Fig. 8: Construction of the ontological subgraph from the definition of *hammer-n03481172*

4.5 Evaluation

In this section, we provide different evaluations of our ontological graph. At the end of the paper figures 19 and 20 provides different subgraphs extracted from our ontology for qualitative evaluation.

	POS		Quantity	Ratio(%)
Node	*Noun*		23945	58.38
	Factor		9338	22.77
	Verb		5517	13.45
	Adjective		2219	5.41
Edge	*isA*	WordNet	21191	20.06
		Auto	15079	14.27
	linkedTo		30379	28.76
	prop		17000	16.09
	homonym		13092	12.39
	hasA		3679	3.48
	usedFor		3020	2.86
	on		2198	2.08

(a) Node/Edge proportions. *isA* relations not originally present in WordNet and extracted by our methodology correspond to the *Auto* line.

Characteristic	*Value*
Mean degree	2.575
Diameter (graph as undirected)	16
Mean distance	5.227

(b) Global characteristics

small, large, form, material, substance, device, part, body, food, surface, structure, fabric, metal, instrument, component, drug, long, building, liquid, side, compound, acid, area, treat, wood, river, object, end, place, various, line, white, make, thin, vein, flesh, use, cell, artifact, ground, head, other, plant, stone, bone, point, source, system, person, several, mixture, arm, mineral, meat, skin, container, light, hold, color

(c) Nodes ordered by *eigenvector centrality*

Table 3: Ontology global description

4.5.1 General graph statistics

We give a global description in the Table 3. First, we see that homonymy relations are far from negligible, confirming the importance of the disambiguation problem. It also shows that a good proportion of extracted hyper/hyponymy relations were not present in the WordNet ontology. Table 3b exposes global characteristics of the ontological graph. We recall that the degree of a node is the number of incident edges. The diameter of a graph is defined as the longest shortest path between two nodes. Finally, Table 3c presents concepts ordered by their *eigenvector centrality* [40] which measures their influences in the graph. As expected, we find rather generic concepts.

4.5.2 Quantitative analysis

It is difficult to quantitatively evaluate the relations built in our ontology. Nevertheless, we manually analyze small randomly chosen samples (200) for each relation type and put the results in the Table 4. The total number of relations (edges) for each type can be found in the Table 3a. Note that for *isA* relations we only use relations extracted by our methodology which were not present in WordNet. Homonym relations are not represented as they are also directly extracted from WordNet. We do not compute accuracies for *linkedTo* relations as they correspond to unknown relations and their pertinence is too subjective to assess. Considering the lack of statiscal significance of our samples set, accuracy is only given on an indicative basis.

Failures can account for wrong or incomplete relations. We see that our approach creates some useful relations. For instance,

$$isA(vinegar\text{-}n07828987, liquid)$$

brings a new description of the vinegar concept lacking in the WordNet ontology where we only have

$$isA(vinegar\text{-}n07828987, condiment\text{-}n07810907)$$

We explain the drop in accuracy for the *hasA* and *on* relations from the lack of some patterns in our approach. Currently, patterns such as *has a/the * of ** are not taken in account and will be integrated in a future version. Some errors come from SyntaxNet which incorrectly considers some adjectives or past participles (*shallow, bent*) as nouns.

We consider that quantitative comparison with other ontologies (KnowRob, NELL, ConceptNet [47]) is not pertinent. Indeed, we are limited to compare them manually which limits us to relatively small samples with no statistical significance. Furthermore, their ontologies are based from different sources which can contain manually defined relations (WordNet, Open Mind common sense [46]) or validated by volunteers. Ultimatly, our objectif is to complement already existent ontologies and use them also to robustify our own methodology.

Relation	Accuracy (95% CI)	Correct examples	Failures
isA	94% ± 3.3	(vinegar-n07828987,liquid) (Tuileries-n04496173,palace) (CD_player-n02988304,equipment) (clabber-n07850219, milk) (dust-n14840092,particule) (paraboloid-n13897002,surface)	(central_nervous_system-n05480794,part) (sublimate-n15062284,product) (main-n09345932,body) (katharometer-n03609147,measure)
hasA	84.5% ± 5.0	(revolver-n04086273,cylinder) (knife-n03623556,handle) (French_dressing-n07833816,mustard) (hammer-n03481172,head) (Emmenthal-n07854982,hole) (marmite-n03722827,leg)	(triangle-n04480853,bent) (mangosteen-n07763987,juicy) (spoon-n04284002,shallow) (Roman_nose-n05599501,prominent) (barbed_wire-n02790823,regular)
prop	96% ± 2.7	(wafer-n07695012,thin) (wheel-n04575723,circular) (sapphirine-n15012810,blue) (jawbreaker-n07599161,hard) (headpiece-n03504205,protective) (dolmen-n03220237,large) (painting-n03876519,artistic)	(loft-n03686470,other) (hydrocarbon-n14911057,only) (synchromesh-n04375241,same) (encephalogram-n03285730,X)
usedFor	94.5% ± 3.2	(drinking_vessel-n03241496,drink) (peavey-n03903133,handle) (clothesbrush-n03050453,clean) (chessboard-n03014317,play) (estradiol-n14750316,treat) (food_processor-n03378174,blend)	(tin-n14658855,alloy) (making-n03714899,perform) (steam_engine-n04309049,raise)
on	85.5% ± 4.9	(wallet-n04548362,money) (mannequin-n03717921,clothes) (butcher_knife-n02927053,meat) (bowl-n02881193,liquid) (parer-n03890093,fruit) (spoon-n04284002,food)	(stealth_fighter-n04308397,bomb) (liqueur_glass-n03676623,amount) (tire_iron-n04441093,shell)

Table 4: Examples of extracted relations taken from our analyzed samples (200 per relation type).

4.6 Discussion

In this section, we introduced the semantic layer of our environment modeling architecture. It is represented as an ontological graph built automatically by analyzing word definitions. Such a hierarchical representation provides some natural benefits in terms of computational complexity. We computed the distance to the root node for each of the $N_{leaf} = 23330$ leaf nodes of our ontological graph and obtained an almost gaussian distribution with a mean distance $\bar{d} = 5$ and a maximum distance $d_{max} = 15$. In particular, we exploit this property in the *Identification* process (Figure 2) which links our perception model to the instance model.

The details of this process are out of the scoop of this paper as they relate to image processing and machine learning. In short, for our architecture to scale with an open world and an unknown number of object classes, we can not use a unique multiclass classifier and have to rely on binary classifiers attached to each concept of the ontological graph. The idea is then to exploit the ontological graph as a decision tree. We reduce then the number of classifiers required to classify an object. It also provides a way to connect unknown visualized objects on intermediate nodes of the ontology. In the following section, we propose another use of its graph structure.

5 Application : notion of context

As explained in the introduction, the ontological graph can be used to extract what we call a *context* of concepts. We propose two definitions of it with slightly different goals :

- Context by co-occurrences of concepts. This definition is the most frequent in practical robotic applications. It is generally used in probabilistic frameworks as a prior information. In an open set world, it could be used to preload classifiers on a limited subset of concepts likely to be present in the current context of the robot. However, it does not give directly what the *context* is.
- Context as the *confounding* nodes explaining the co-occurrences of concepts. This can be used to infer new relations between concepts through their common context. This is the example of *fork* and *restaurant* in Figure 9.

In the following, we investigate measures for each of those definitions. It is rather difficult to evaluate such methods. Indeed, there is no available ground truth and more importantly the results are dependent on the ontological graph used. We evaluate it here empirically through examples. In the case of a unique concept, we use its centered subgraph. For a subset of concepts, we use the union of their subgraphs.

5.1 Measure of context based on word vector representation

A simple way to have a measure of *context* is to use word vector representation such as GloVe [35]. Those representations learns words embedding in a vector space

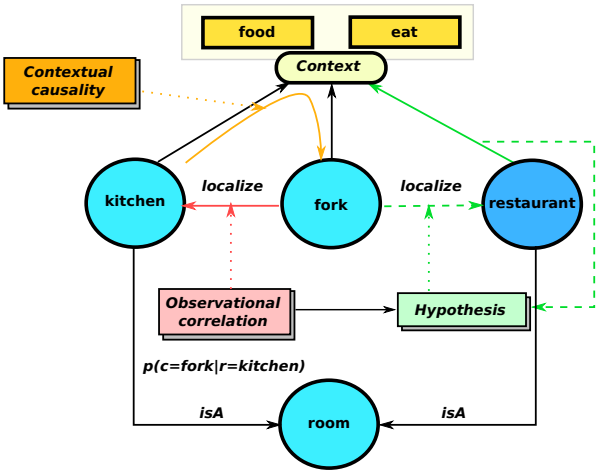


Fig. 9: Illustration of the notion of "context" as defined in our work, acting similarly to confounding variables

based on textual corpus. As they are based on local neighbourhood, they capture the context through co-occurrences of words. The measure of context p_{GloVe} between two nodes c_i, c_j is the cosine similarity of their corresponding word vector $G(c_i), G(c_j)$:

$$p_{GloVe}(c_i, c_j) = \frac{G(c_i)^T G(c_j)}{||G(c_i)||_2 ||G(c_j)||_2} \tag{2}$$

When no vector representation is found for a concept, we use instead its first hypernyms with an available representation. This simple method returns an ordered list of related concepts which can be present in the same scene as the targeted concept. Two concepts with a high similarity are likely to share a common context. However, it does not generally imply that a concept has high similarity with concepts representing its context. For instance, *fork* and *spoon* have a rather high similarity ($p_{GloVe}(fork, spoon) = 0.44$) whereas their similarities with *food* is lower ($p_{GloVe}(fork, food) = 0.08, p_{GloVe}(spoon, food) = 0.11$).

Although its usefulness in practice, this measure has some drawbacks. First, GloVe is a unimodal representation. This means that homonyms are represented by the same feature vector. Furthermore, this approach can't take in account abstracted or action-related context. This can be explained by the fact that the words *fork* and *food* do not *explicitly* appear together in the training corpus. To extract such information, we have to rely on the ontological graph structure.

5.2 Measure of context based on betweenness centrality

Intuitively, context concepts are nodes with a high centrality value in the subgraph of the targeted concept (as *food* in figures 4e and 4f). In graph theory, there are several definitions of centrality. As context concepts are expected to be confounders between various concepts, we propose to use a slightly modified version of betweenness centrality [16][6]. It measures the proportion of a node to be in the shortest path between two other nodes. Formally, for a graph $G = \{V, E\}$, the betweenness centrality bc of a node $v \in V$ is given by

$$bc(v) = \sum_{s,t \in V, s \neq\ t \neq v} \frac{\sigma(s, t|v)}{\sigma(s, t)} \tag{3}$$

where $\sigma(s, t)$ is the total number of shortest path between two nodes s and t. $\sigma(s, t|v)$ is the number of shortest path going through v. The result can be normalized by the number of pairs of nodes. However, we can't use this measure directly in our graph :

- A high value of betweenness centrality is a necessary but not a sufficient condition : nodes with high degree are prone to high betweenness centralities.
- We have to adapt the measure when dealing with factor nodes. They should not be taken in account and share their centrality values to their direct neighbourhood.

Those two problems are addressed by the following modification to the original measure (2). First, we normalize the betweenness centrality value by the degree of the node :

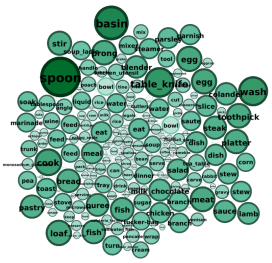
$$bc'(v) = \frac{1}{deg(v)} \sum_{s,t \in V, s \neq\ t \neq v} \frac{\sigma(s, t|v)}{\sigma(s, t)} \tag{4}$$

In addition, we remove node with a degree 1 not related to a factor node : they don't provide information on context. Let $v_f \in V_K^F$ be a factor node of our ontological graph and $C(v_f)$ the set of its neighbour nodes. The modified betweenness centrality bc' is given by :

$$\forall v \in C(v_f),\ bc'(v) = bc(v) + \frac{bc(v_f)}{deg(v_f)} \tag{5}$$

$$bc'(v_f) = 0 \tag{6}$$

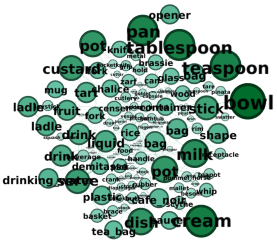
The figures 10a,10b show examples with node size proportional to the GloVe similarity measure and our normalized version of betweenness centrality. More examples are provided at the end of the paper in Figure 21.



(a) GloVe similarities for *fork-n03383948*

(b) Betweenness centralities for *fork-n03383948*

(c) GloVe similarities for the subset *cup-n03147509, spoon-n04284002*



(d) Betweenness centralities for the subset *cup-n03147509,spoon-n04284002*

Fig. 10: Examples of context relatedness measures as exposed in section 5. Nodes size and color are proportional to their contextual measure values

5.3 Extension to common context

The previous sections deal with the context of one concept. Another interesting case is its extension to the *common context* of a set of concepts. Let $\mathcal{C} = \{c_i\}_{1,\dots,m}$ a set of concepts. We consider the subgraph $G_{\mathcal{C}}$ defined as the union of subgraph G_{c_i} centered on each $c_i \in \mathcal{C}$:

$$G_{\mathcal{C}} = \bigcup_{i=1}^{m} G_{c_i} \tag{7}$$

In the case of the betweenness centrality, the extension is immediate by applying the measure on the subgraph G_C. For the approach based on GloVe vector representation, we need to adapt the measure. By noting $P(E)$ the set of subset of a set E, we define a new context measure $p^c_{GloVe} : P(V_k) \times V_K \mapsto [0, 1]$ as the geometric mean of the similarities between a concept $c \in G_{\mathcal{C}}$ to each concept of $\mathcal{C}$:

$$p^c_{GloVe}(\mathcal{C}, c) = \sqrt[m]{\prod_{i=1}^{m} p_{GloVe}(c_i, c)} \tag{8}$$

Example for the subset *cup-n*03147509, *spoon-n*04284002 is given in Figure 10d. Geometric mean was chosen here because it decreases as the dispersion in the similarities $p_{GloVe}(c_i, c)$ increases.

5.4 Discussion

The measure of context based on the ontology structure (betweenness centrality) is a first step towards *transfert learning* of facts learned from observational correlations as illustrated in Figure 9. Figure 21 gives some more examples at the end of the paper. We provide a proof-of-concept through qualitative evaluation. The measure based on GloVe similarity is more practical as it allows, in our open-set world assumption, to give priorities to objects sharing the current context of the robot. Preliminary works using current flow betweenness centrality [7] seems promising by giving more robust evaluation of context as it is not limited to shortest path between concepts.

6 Human-robot interactions : task creation from request

This section deals with how our system process informations coming from human users. We mainly consider three type of information :

- **General information** eg. *A cup has a handle.* Such information is processed as explained in section 4.4 and added to the ontological graph.
- **Specific information** concerning instances eg. *My cup is red.* Semantic analysis is also similar to the previous point. The corresponding instance is created or updated in the instance model. For instance, the input *My cup is red* made by a user $userName$ translates to a Prolog rule *my_cup/1* :

$$\begin{aligned} my\_cup(A) :\text{-}\ & isA(A, cup), prop(A, red), \\ & belongTo(A, userName). \end{aligned} \tag{9}$$

  where $belongTo$ is a specific predicate used in the instance model. In fact, as ownership can't be detected by observation, we separate the instance "core" definition and create a new rule for ownership so that (9) is replaced by :

$$my\_cup(A) :\text{-}\ isA(A, cup), prop(A, red). \tag{10}$$

$$belongToUserName(A) :\text{-}\ my\_cup(A). \tag{11}$$

- **Request** eg. *Bring me my cup.* There are converted into Prolog rules. Solutions are searched by exploiting both perception (instances detections), instances definitions (if any) and generic knowledge (ontology).

In the following, we focus on the request case.

Table 5: Decomposition of the request *Bring me the fork with a red handle behind my small cup* into prolog clauses. We use *t.w.f.a.r.h* to refer to *the fork with a red handle*

Clause	**P**	$\boldsymbol{\alpha}^i$	$\boldsymbol{\alpha}^c$	$\boldsymbol{\alpha}^f$
isA(t.f.w.a.r.h, fork)	isA	{t.f.w.a.r.h}	{fork}	$\emptyset$
isA(my_small_cup, cup)	isA	{my_small_cup}	{cup}	$\emptyset$
isA(red_handle, handle)	isA	$\emptyset$	{handle}	{red_handle}
prop(red_handle, red)	prop	$\emptyset$	{red}	{red_handle}
hasA(t.f.w.a.r.h, red_handle)	hasA	{t.f.w.a.r.h}	$\emptyset$	{red_handle}
prop(my_small_cup, small)	prop	{my_small_cup}	{small}	$\emptyset$
behind(the_fork, my_cup)	behind	{the_fork, my_cup}	$\emptyset$	$\emptyset$

6.1 Request conversion in Prolog rule

6.1.1 Prolog clause definition

First, in our work we define a Prolog clause F as

$$F = P\left(\boldsymbol{\alpha} = \{a_1, \ldots, a_n\}\right)[\boldsymbol{\alpha}^i,\ \boldsymbol{\alpha}^c,\ \boldsymbol{\alpha}^f] \tag{12}$$

where P is the predicate and $\boldsymbol{\alpha}$ its set of arguments. Note that in Prolog, constants (atomic terms) start with a lower case and variables with a upper case. $\boldsymbol{\alpha}^i$ (resp. $\boldsymbol{\alpha}^c$ and $\boldsymbol{\alpha}^f$) are the subsets of arguments corresponding to instances (resp. generic classes (concept) and intermediate equivalent to factor nodes). Those clauses are extracted from the request similarly to the section 4.4. An example of a decomposition following the definition (12) is given in Table 5. Predicates are the same as in the ontological graph with addition of spatial relations (*behind,to the left of,...*) and existence (*isDetected*). We use here the concept of "computable classes" in KnowRob [51] where some predicates are checked at run-time based on perception.

6.1.2 Prolog rule conversion

Formally we represent the request R_m/k as a Prolog rule based on the Prolog clause decomposition :

$$R_m(\mathbf{A})[\boldsymbol{\alpha}^i_m, \boldsymbol{\alpha}^c_m, \boldsymbol{\alpha}^f_m] :\text{-}\ \bigwedge_{j=1}^{N} F_j(\sigma(\boldsymbol{\alpha}_j))[\boldsymbol{\alpha}^i_j,\ \boldsymbol{\alpha}^c_j, \boldsymbol{\alpha}^f_j]. \tag{13}$$

$\mathbf{A} = \{A_1, \ldots, A_k\}$ is a set of variable arguments which can correspond to concepts (eg. *cup*) or instances (eg. *my cup*). We have the mapping

$$\sigma : \bigcup_{j=1}^{N} \boldsymbol{\alpha_j} \rightarrow \bigcup_{j=1}^{N} \boldsymbol{\alpha_j} \cup \mathbf{A} \tag{14}$$

which sends some arguments of the clauses F_j to the variable arguments $\mathbf{A}$. We also define for $t \in \{i, c, f\}$

$$\boldsymbol{\alpha}^t_m = \bigcup_{j=1}^{N} \boldsymbol{\alpha}^t_j \tag{15}$$

respectively the union of the instance, concept and factor constants from the clause arguments. Generally predicates are directly obtained from the syntactical analysis of the request. However, clauses involved with instance arguments generate different predicates depending on the situation. For the sake of clarity, consider the simple request *Bring me my cup behind the fork*. It translates to the following Prolog rules :

- In case the instance (here *my cup*) has already be seen[2], we have :

$$R_m(A_1) :\text{-}\ isDetected(my\_cup),\ isA(A_1, fork),\ behind(my\_cup, A_1). \tag{16}$$

  with $\sigma(the\_fork \in \boldsymbol{\alpha}^i_m) = A_1$.
- In case the instance has not been seen but is already semantically defined, we have :

$$R_m(A_1, A_2) :\text{-}\ my\_cup(A_2),\ isA(A_1, fork),\ behind(A_2, A_1). \tag{17}$$

  where $my\_cup/1$ is the corresponding Prolog rule (as in (10)). Here

$$\sigma(the\_fork \in \boldsymbol{\alpha}^i_m) = A_1, \sigma(my\_cup \in \boldsymbol{\alpha}^i_m) = A_2$$

- In the remaining case where the instance is unknown, we have :

$$R_m(A_1, A_2) :\text{-}\ isA(A_1, fork),\ isA(A_2, cup),\ behind(A_2, A_1). \tag{18}$$

  with the same mapping σ as in the previous case.

A detection thread is started for each concept $c \in \boldsymbol{\alpha}^c_m$ and instance $i \in \boldsymbol{\alpha}^c_m$ appearing in the Prolog rule. In both case, we use a pretrained Convolutional Neural Network [20, 45, 50] casted to a Fully Convolutional Network (FCN) [26] for detection. We then replace the last softmax layer respectively by a binary Random Forest layer or a cosine similarity layer as illustrated in Figure 11.

[2] In other words, it means that it can be detected by the perception module

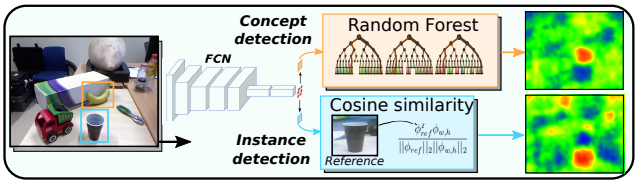


Fig. 11: Modified version of FCN used in our work. Binary Random Forest are used for concept detection. Cosine similarity with previous observations is used for instance detection

7 Use cases

An intrinsic evaluation of our model is hard to obtain as the results rely heavily on external tools (CNN, SyntaxNet). In order to assess the validity of our approach, we propose to illustrate our system through some use cases.

7.1 Experimental settings

We consider a scene consisting of several table-top objects. We use a RGBD Kinect sensor and detect the table plane using a RANSAC-based plane fitting on the 3D point cloud. Objects above the table are clustered and then registered as instances. The initial 2D object masks are further refined using GrabCut algorithm [39]. The system core is implemented in C++. Prolog is integrated through the C/C++ interface of the SWI-Prolog 7.7.12 environment [56]. FCNs for concept and instance detection are implemented using Caffe framework [18] based on AlexNet [20] and VGGNet [45]. We use the Random Forest implementation of OpenCV and train those binary classifiers with the image database ImageNet [15].

7.2 Multi-instances scenario

We consider an initially empty scene. The user (here *Yohan*) provides some specific information to the system : *"My cup is to the left of the car"*. From the syntactical analysis, the semantic definition of the instance *my_cup* induced Prolog rules as in figure 12 :

$$my\_cup(A) :- \ isA(A, cup),\ isA(B, car),\ to\_the\_left\_of(A, B). \quad (19)$$
$$belongToYohan(A) :- \ my\_cup(A).$$

Objects are then added to the scene and passively detected using the method described in section 7.1 (Figure 13). Each detection generates a new instance in the instance model. The request *"Bring me the cup behind*

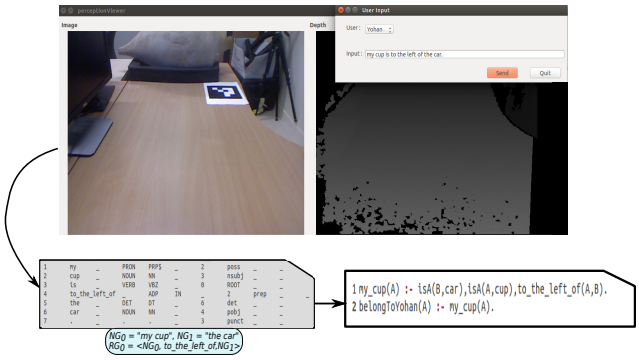


Fig. 12: Specific information provided by the user generates an instance defined semantically

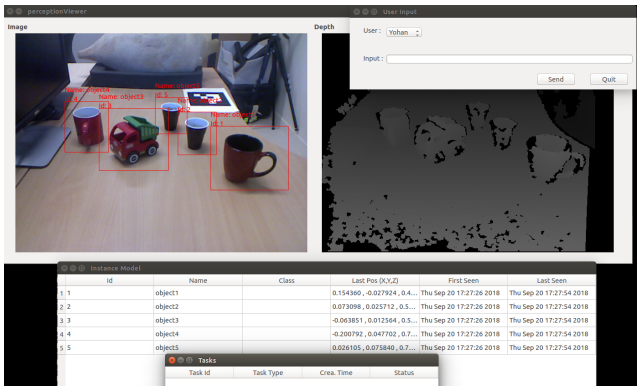


Fig. 13: State before the request after introduction of objects

my cup" is then made by the user. Following section 6.1.2, it is internally represented by the rule

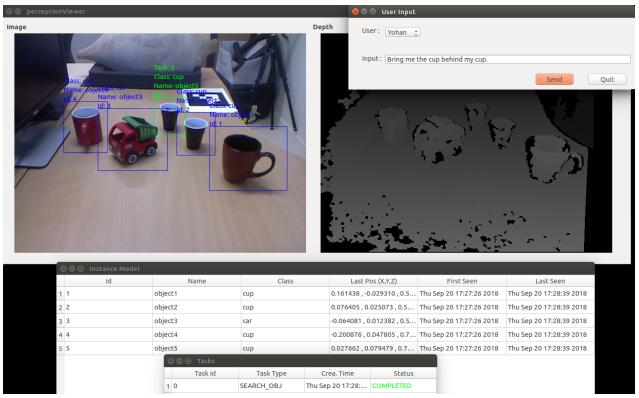
$$R_m(A, B) :- \ isA(A, cup), my\_cup(B), behind(A, B). \quad (20)$$

Two threads for the detection of *cup* and *car* (required by the *my_cup* clause) are started. Spatial clauses related to current observations are evaluated on the fly and stored in a temporary file.

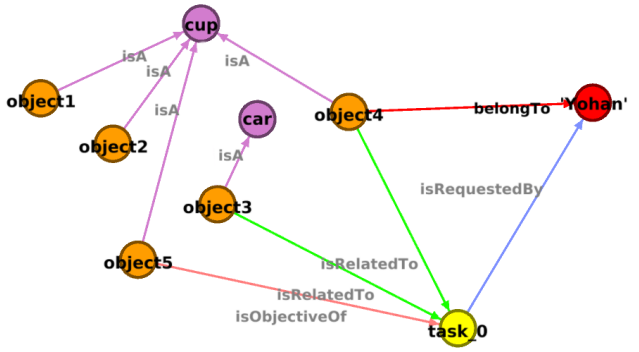
Finally, the task is completed by finding the requested instance (figure 14a). The instance model is thus updated as in figure 14b.

7.3 Ontology usage and opportunist detection

We propose a case involving directly our generated ontology. The scene can be seen in figure 16. Note that the spoon has not been detected because it was too close to the Kinect sensor. We consider the following request : *"Give me something for serving coffee"*. Unlike the previous case, the requested object is not directly given. Here our ontology provides the subset of



(a) Final state of the system



(b) Updated instance model after task completion

Fig. 14: Final state after the request resolution

candidate concepts. After the syntactical analysis of the request, we obtain the clause which should be satisfied by the requested object as well as the rule representing the task

$$conceptDef(A) :\text{-}\ \ useFor(A,F), on(F,coffee), \\ isA(F,serve). \qquad (21)$$

$$R_m(A) :\text{-}\ conceptDef(B),\ isA(A,B). \qquad (22)$$

where F is a variable corresponding to a factor node.

First, the system searchs for the sets of concepts (synset) corresponding to the words *coffee* and *serve*. We obtain *coffee-n07929519* and *serve-v01181295/ v01180351/v01428011/v01438681*. For each pair of concepts, we search through the ontology using the rule (21) as shown in Figure 15. The negative integers are the identifiers used for factor nodes. Two concepts are satisfying (21) : *demitasse-n03174731* and *coffee_mug-n03063599*. Each of those concepts are searched in separate threads similarly to the previous case. Finally, an instance is found to be a *coffee_mug-n03063599* which solves (21) (figure 16). At this point, the system knows that the scene is composed of a *coffee_mug*. One opportunist strategy to detect new instances in the scene is to exploit the current *context* from the instance model.

```
?- useFor(A,B),actOn(B,coffee-n07929519),isA(B,serve-v01181295).
A = demitasse-n03174731,
B = -1303 ;
A = coffee_mug-n03063599,
B = -1012 ;
false.

?- useFor(A,B),actOn(B,coffee-n07929519),isA(B,serve-v01180351).
false.

?- useFor(A,B),actOn(B,coffee-n07929519),isA(B,serve-v01428011).
false.

?- useFor(A,B),actOn(B,coffee-n07929519),isA(B,serve-v01438681).
false.

?-
```

Fig. 15: Search of requested concept with the Prolog engine

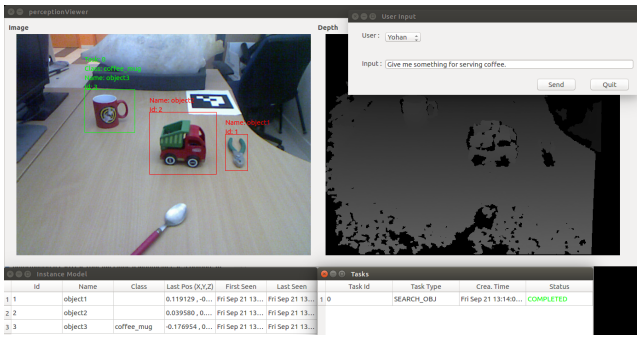


Fig. 16: System after task completion in the use case illustrating ontology usage and opportunist detection

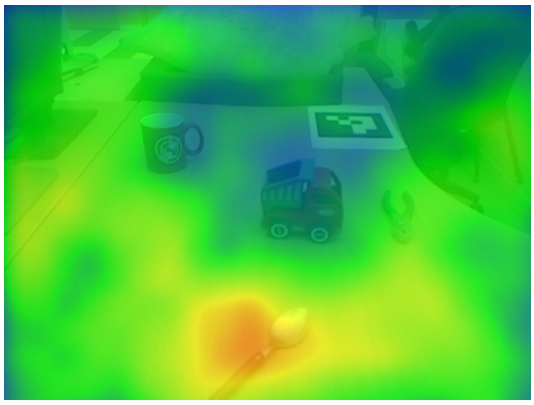
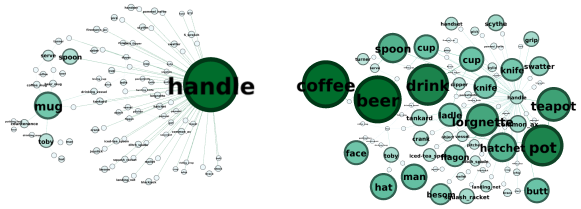


Fig. 17: Heatmap obtained from the opportunist detection of a spoon using context information

Figure 18 shows the ontological subgraph of *coffee_mug* with node size proportional to the context measure employed. The first concepts are then searched for with a Random Forest FCN (Figure 11). Figure 17 shows the resulting heat map for the spoon concept.

7.4 Discussion

In this section we illustrate how our system globally works on simple usage cases. This practical implementation allowed us to validate our approach. Note that



(a) Betweenness centrality measure (b) GloVe similarity measure

Fig. 18: Context for *coffee_mug-n03063599* with GloVe similarity and betweenness centrality

the objective is to illustrate the ideas and architecture presented throughout this paper. In a real settings, it would be necessary to optimize the different algorithms used and notably the binary Random Forest classifiers. Although it is not refined enough yet for real-world exploitation, this proof-of-concept gives encouraging results and encourage us to pursue our efforts in this direction.

8 Conclusion and Future work

The vast majority of the robotic literature focuses on task description. While we are convinced that the task-oriented approach is fundamental, little has been done on environment representation for its own sake. It is generally defined at hand according to the targeted task. The originality of our approach is to uncorrelate (as possible) the environment representation from the final task. We propose a general, task-independent environment representation for robotic applications where interactions with human are involved. It encompasses classical representations (geometrical, instance-oriented, semantic) in a unique framework.

In particular, this paper is focused on the semantic knowledge representation and human-robot interactions. Knowledge representation is often done using ontologies. From a robotics point of view, they often concerns unrelated domains (eg. History, Politics) and in the same time lacks detailed description of usual physical objects. We propose an original method to generate automatically an object-based ontology from WordNet definitions where concepts are defined by their physical appearance and/or their functions (usage). We validate our approach by visually inspecting the generated graph and by evaluating samples of inferred relations.

Based on the ontological graph, we make a first step towards the formalization of what we usually call *context*. We proposed two main measures of *context* relatedness and empirically showed their pertinence.

Finally, we discussed about how our system deals with human interactions. We propose a formalization for request conversion into Prolog rules. This is validated by two simple use cases which illustrates how our system can be used.

There is still a lot of room for improvements. As our measure of context is directly related to the quality of our ontology, we are first considering a refined process for the ontology generation. The idea is to recursively analyse WordNet definitions using the ontology built in the previous iteration and learns new predicates/rules for relation extraction. We also envisage to apply the method to different dictionaries and merge the resulting graphs. Another possibility is to filter pertinent relations from existing ontologies and to integrate them. In particular, ConceptNet [47] proposes relations similar to ours and is a good candidate for integration.

Feedbacks between the robot and users are an absolute necessity when requests have no or several solutions. In both cases, the robot should think with contrafactualities : think about what could have been if but is not (from its point of view). Our perspectives include the study of an approach for generating questions (feedback) in order to obtain a unique solution to the request.

Note that our work shows some similarities with the approach proposed by [11] and [34]. Indeed, current AI trend of deep machine learning is to learn correlations from data. We believe that the AI field (and autonomous robotics) should leverage *causality* in order to adapt and learn in an open world. However, those two approaches are not mutually exclusive. The recent AlphaGo Zero deep reinforcement learning system [44] for playing Go has demonstrated impressive results without any external data but the rules of the game. Our work can be thought as a step towards defining *rules* describing a robot environment. Its use in a reinforcement learning framework is a promising research path.

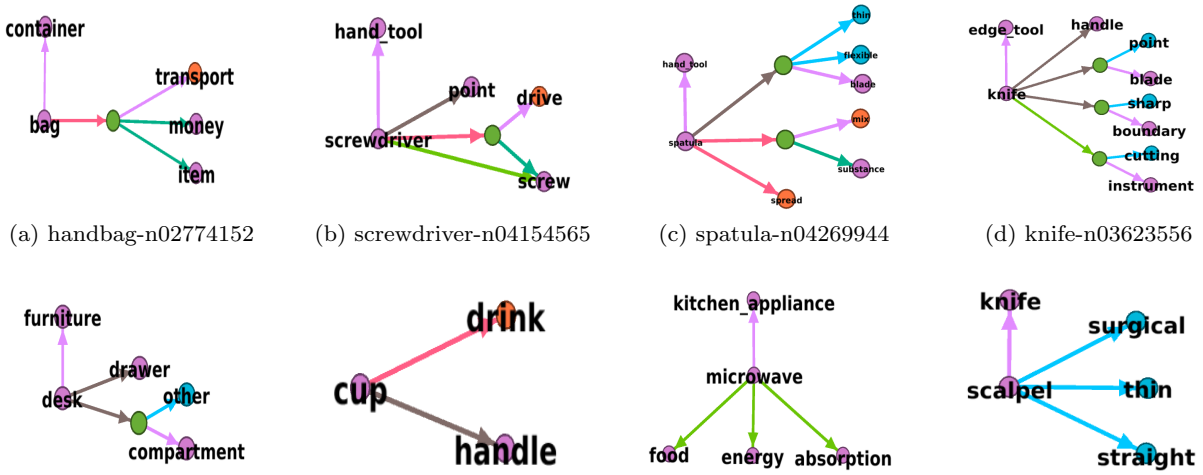
Conflict of Interest : The authors declare that they have no conflict of interest.

References

1. Andor D, Alberti C, Weiss D, Severyn A, Presta A, Ganchev K, Petrov S, Collins M (2016) Globally normalized transition-based neural networks. arXiv preprint arXiv:160306042
2. Auer S, Bizer C, Kobilarov G, Lehmann J, Cyganiak R, Ives Z (2007) Dbpedia: A nucleus for a web of open data. In: The semantic web, Springer, pp 722–735
3. Belpaeme T, Kennedy J, Ramachandran A, Scassellati B, Tanaka F (2018) Social robots for education: A review. Science Robotics 3(21):eaat5954

4. Ben Amor H, Neumann G, Kamthe S, Kroemer O, Peters J (2014) Interaction primitives for human-robot cooperation tasks. In: 2014 IEEE International Conference on Robotics and Automation (ICRA), pp 2831–2837, DOI 10.1109/ICRA.2014.6907265
5. Bracewell DB, Ren F, Kuroiwa S (2006) Towards knowledge about causal agents in wordnet. In: Proceedings of the 10th WSEAS international conference on Computers, World Scientific and Engineering Academy and Society (WSEAS), pp 564–568
6. Brandes U (2001) A faster algorithm for betweenness centrality. Journal of mathematical sociology 25(2):163–177
7. Brandes U, Fleischer D (2005) Centrality measures based on current flow. In: Annual symposium on theoretical aspects of computer science, Springer, pp 533–544
8. Breux Y, Druon S, Zapata R (2018) From perception to semantics: An environment representation model based on human-robot interactions. In: 2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), IEEE, pp 672–677
9. Buchholz S, Marsi E (2006) Conll-x shared task on multilingual dependency parsing. In: Proceedings of the Tenth Conference on Computational Natural Language Learning, pp 149–164
10. Carlson A, Betteridge J, Kisiel B, Settles B, Jr ERH, Mitchell TM (2010) Toward an architecture for never-ending language learning. In: Proceedings of the Twenty-Fourth Conference on Artificial Intelligence (AAAI)
11. Darwiche A (2018) Human-level intelligence or animal-like abilities ? Communications of the ACM 61(10):56–67
12. De Boni M, Manandhar S (2002) Automated discovery of telic relations for wordnet. In: Proceedings of the first International WordNet conference
13. De Marneffe MC, Manning CD (2008) Stanford typed dependencies manual. Tech. rep., Technical report, Stanford University
14. De Marneffe MC, Dozat T, Silveira N, Haverinen K, Ginter F, Nivre J, Manning CD (2014) Universal stanford dependencies: A cross-linguistic typology. In: LREC, vol 14, pp 4585–4592
15. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009) Imagenet: A large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 248–255
16. Freeman LC (1977) A set of measures of centrality based on betweenness. Sociometry pp 35–41
17. Frennert S, Eftring H, Östlund B (2017) Case report: Implications of doing research on socially assistive robots in real homes. International Journal of Social Robotics 9(3):401–415, DOI 10.1007/s12369-017-0396-9, URL https://doi.org/10.1007/s12369-017-0396-9
18. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, Guadarrama S, Darrell T (2014) Caffe: Convolutional architecture for fast feature embedding. arXiv preprint arXiv:14085093
19. Kennedy J, Baxter P, Senft E, Belpaeme T (2016) Social robot tutoring for child second language learning. In: The Eleventh ACM/IEEE International Conference on Human Robot Interaction, IEEE Press, pp 231–238
20. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
21. Lang D, Paulus D (2014) Semantic maps for robotics. Proc of the Workshop "Workshop on AI Robotics" at ICRA
22. Lang D, Friedmann S, Häselich M, Paulus D (2014) Definition of semantic maps for outdoor robotic tasks. In: IEEE International Conference on Robotics and Biomimetics, pp 2547–2552
23. Lang D, Friedmann S, Hedrich J, Paulus D (2015) Semantic mapping for mobile outdoor robots. In: 14th IAPR International Conference on Machine Vision Applications, pp 325–328
24. Leite I, Martinho C, Paiva A (2013) Social robots for long-term interaction: A survey. International Journal of Social Robotics 5(2):291–308, DOI 10.1007/s12369-013-0178-y, URL https://doi.org/10.1007/s12369-013-0178-y
25. Lenat DB (1995) Cyc: A large-scale investment in knowledge infrastructure. Communications of the ACM 38(11):33–38
26. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 3431–3440
27. Matuszek C, Cabral J, Witbrock MJ, DeOliveira J (2006) 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, pp 44–49
28. Miller GA (1995) Wordnet: a lexical database for english. Communications of the ACM 38(11):39–41
29. Mitchell T, Cohen W, Hruschka E, Talukdar P, Betteridge J, Carlson A, Dalvi B, Gardner M, Kisiel B, Krishnamurthy J, Lao N, Mazaitis K, Mohamed

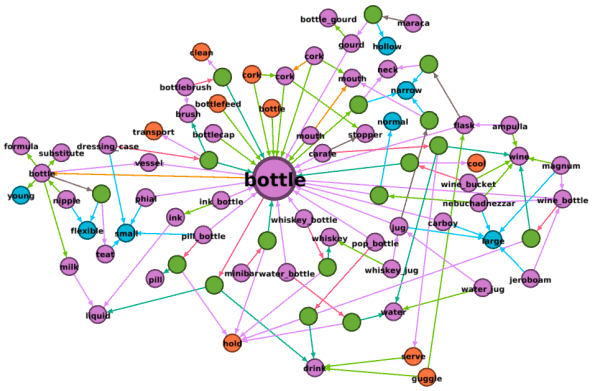
T, Nakashole N, Platanios E, Ritter A, Samadi M, Settles B, Wang R, Wijaya D, Gupta A, Chen X, Saparov A, Greaves M, Welling J (2015) Never-ending learning. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence
30. Mukai T, Hirano S, Nakashima H, Kato Y, Sakaida Y, Guo S, Shigeyuki H (2010) Development of a nursing-care assistant robot riba that can lift a human in its arms. pp 5996 – 6001, DOI 10.1109/IROS.2010.5651735
31. Mur-Artal R, Tardós JD (2017) Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. IEEE Transactions on Robotics 33(5):1255–1262
32. Nivre J, De Marneffe MC, Ginter F, Goldberg Y, Hajic J, Manning CD, McDonald RT, Petrov S, Pyysalo S, Silveira N, et al. (2016) Universal dependencies v1: A multilingual treebank collection. In: LREC, URL http://universaldependencies.org/.
33. Novischi A (2002) Accurate semantic annotations via pattern matching. In: FLAIRS Conference, pp 375–379
34. Pearl J, Mackenzie D (2018) The Book of Why: The New Science of Cause and Effect. Basic Books
35. Pennington J, Socher R, Manning CD (2014) Glove: Global vectors for word representation. In: EMNLP, vol 14, pp 1532–43
36. Petrov S, Das D, McDonald R (2011) A universal part-of-speech tagset. arXiv preprint arXiv:11042086
37. Pronobis A, Jensfelt P (2011) Hierarchical multi-modal place categorization. In: ECMR, pp 159–164
38. Pronobis A, Jensfelt P (2012) Large-scale semantic mapping and reasoning with heterogeneous modalities. In: IEEE International Conference on Robotics and Automation, pp 3515–3522
39. Rother C, Kolmogorov V, Blake A (2004) Grabcut: Interactive foreground extraction using iterated graph cuts. In: ACM transactions on graphics (TOG), ACM, vol 23, pp 309–314
40. Ruhnau B (2000) Eigenvector-centrality ? a node-centrality? Social networks 22(4):357–365
41. Sabelli AM, Kanda T, Hagita N (2011) A conversational robot in an elderly care center: An ethnographic study. In: 2011 6th ACM/IEEE International Conference on Human-Robot Interaction (HRI), pp 37–44, DOI 10.1145/1957656.1957669
42. Santorini B (1990) Part-of-speech tagging guidelines for the penn treebank project (3rd revision). Technical Reports (CIS) p 570, URL http://www.clips.ua.ac.be/pages/MBSP-tags
43. Sciutti A, Bisio A, Nori F, Metta G, Fadiga L, Pozzo T, Sandini G (2012) Measuring human-robot interaction through motor resonance. International Journal of Social Robotics 4(3):223–234
44. Silver D, Schrittwieser J, Simonyan K, Antonoglou I, Huang A, Guez A, Hubert T, Baker L, Lai M, Bolton A, et al. (2017) Mastering the game of go without human knowledge. Nature 550(7676):354
45. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:14091556
46. Singh P, Lin T, Mueller ET, Lim G, Perkins T, Zhu WL (2002) Open mind common sense: Knowledge acquisition from the general public. In: OTM Confederated International Conferences On the Move to Meaningful Internet Systems, Springer, pp 1223–1237
47. Speer R, Havasi C (2012) Representing general relational knowledge in conceptnet 5. In: LREC, pp 3679–3686
48. Sünderhauf N, Dayoub F, McMahon S, Talbot B, Schulz R, Corke P, Wyeth G, Upcroft B, Milford M (2016) Place categorization and semantic mapping on a mobile robot. In: IEEE International Conference on Robotics and Automation, pp 5729–5736
49. Sünderhauf N, Pham T, Latif Y, Milford M, Reid ID (2017) Meaningful maps with object-oriented semantic mapping. In: Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on, IEEE, pp 5079–5085, URL http://arxiv.org/abs/1609.07849
50. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A, et al. (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1–9
51. Tenorth M (2011) Knowledge processing for autonomous robots. PhD thesis, Technische Universität München
52. Tenorth M, Beetz M (2017) Representations for robot knowledge in the knowrob framework. Artificial Intelligence 247:151–169
53. Thrun S, Burgard W, Fox D (2005) Probabilistic Robotics. MIT Press, Cambridge
54. Wada K, Shibata T (2007) Living with seal robots?its sociopsychological and physiological influences on the elderly at a care house. IEEE transactions on robotics 23(5):972–980
55. Whelan T, Leutenegger S, Salas-Moreno RF, Glocker B, Davison AJ (2015) Elasticfusion: Dense slam without a pose graph. Proc Robotics: Science and Systems, Rome, Italy
56. Wielemaker J, Schrijvers T, Triska M, Lager T (2012) SWI-Prolog. Theory and Practice of Logic Programming 12(1-2):67–96



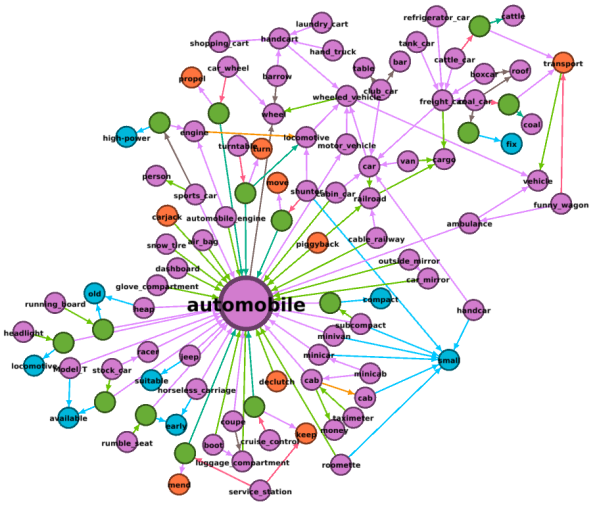
(a) handbag-n02774152 (b) screwdriver-n04154565 (c) spatula-n04269944 (d) knife-n03623556

(e) desk-n03179701 (f) cup-n03147509 (g) microwave-n03761084 (h) scalpel-n02774152

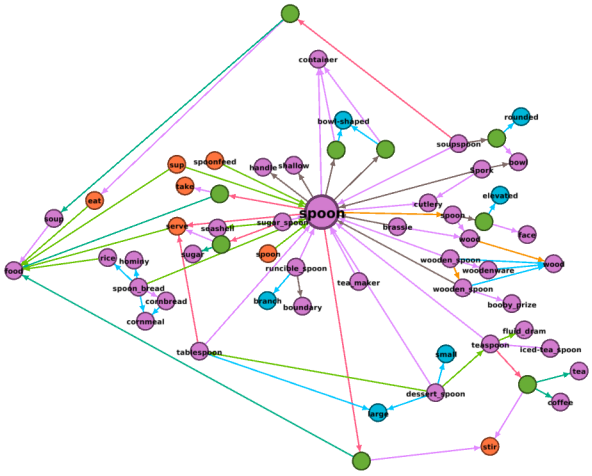
Fig. 19: Examples of subgraphs built from WordNet definitions following the approach explained in section 4.4 (best viewed in color)



(a) bottle-n02876657

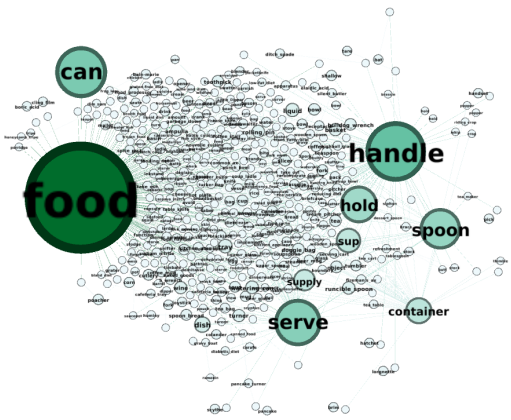


(b) automobile-n02958343

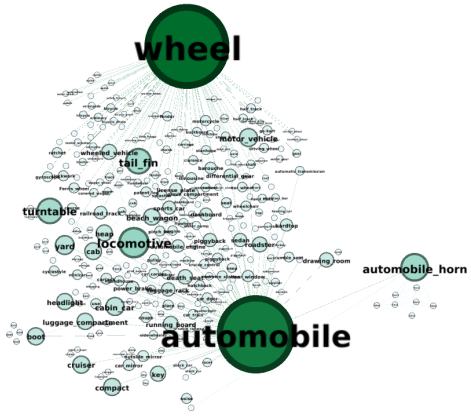


(c) spoon-n04284002

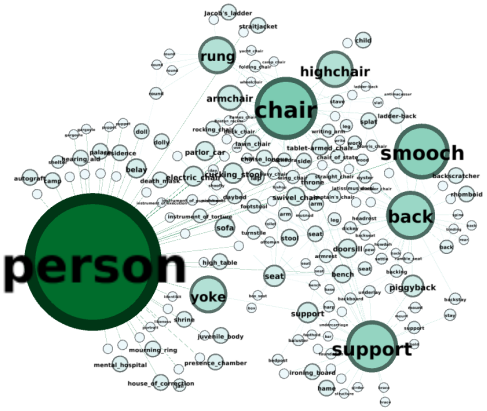
Fig. 20: Subgraphs consisting of nodes at a distance 1 (excluding factor nodes) of a central concept and its hyponyms (best viewed in color)



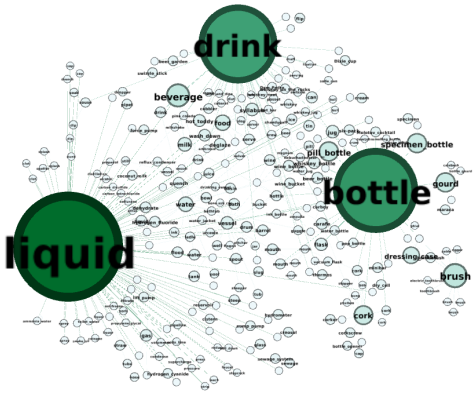
(a) spoon-n04284002



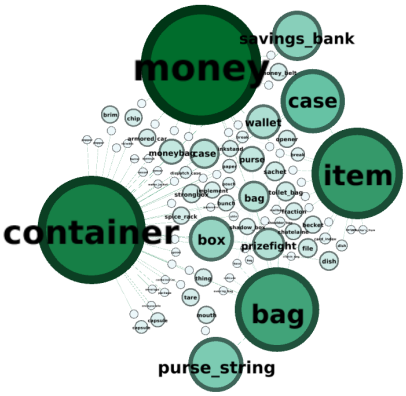
(b) automobile-n02958343



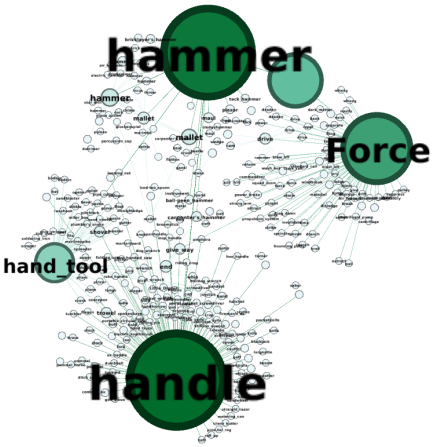
(c) chair-n03001627



(d) bottle-n02876657



(e) handbag-n02774152



(f) hammer-n03481172

Fig. 21: Subgraphs with nodes size and color proportional to their modified betweenness centrality measures