Transforming Web Knowledge into Actionable Knowledge Graphs for Robot Manipulation Tasks

Michael Beetz[1], Philipp Cimiano[2], Michaela Kümpel[1], Enrico Motta[3], Ilaria Tiddi[4] and Jan-Philipp Töberg[2]

[1]*Institute for Artificial Intelligence, University of Bremen, Bremen, Germany*
[2]*Cluster of Excellence Cognitive Interaction Technology (CITEC), Bielefeld University, Bielefeld, Germany*
[3]*Knowledge Media Institute, The Open University, Milton Keynes, United Kingdom*
[4]*Knowledge Representation and Reasoning Group, Vrije Universiteit Amsterdam, The Netherlands*

Abstract

One of the visions in AI based robotics are household robots that can autonomously handle a variety of meal preparation tasks. Based on this scenario, we present a best practice tutorial on how to create actionable knowledge graphs that a robot can use for execution of task variations of cutting actions. We implemented a solution for this task that integrates all necessary software components in the framework of the robot control process. In the context of this tutorial, we focus on knowledge acquisition, knowledge representation and reasoning, and simulating robot action execution, bringing these components together into a learning environment that – in the extended version – introduces the whole control process of Cognitive Robotics. In particular, the Tutorial will detail necessary concepts a knowledge graph should include for robot action execution, how web knowledge can be automatically acquired for the domain of cutting fruits, and how the created knowledge graph can be used to let robots execute tasks like slicing a cucumber or quartering an apple. The learning environment follows an immersive approach, using a physics-based simulation environment for visualization purposes that helps to illustrate the concepts taught in the tutorial.

Tutorial ressource: https://github.com/Food-Ninja/Tutorial_ESWC_HHAI

Keywords

Knowledge Representation, Cognitive Robotics, Web Knowledge, Actionable Knowledge, Knowledge Extraction

1. Introduction

We envision household robots that can be placed in any kitchen to then be given a random recipe from the Web that they can understand and parse into action plans that can be broken down into executable body motions that can be performed with available objects in the environment. For this, robots need to be enabled to perform meal preparation tasks with any tool, on any available object and for a variation of tasks. This tutorial is based on prior research that proposed a methodology for creating actionable knowledge graphs [1], where a solution for creating knowledge graphs that link object to action and environment information and thus make them *actionable* is proposed, as well as a knowledge engineering methodology that is more specifically aligned to creating ontologies for meal preparation tasks that can be used to parameterise robot action plans in order to perform task variations of cutting actions [2].

There has been lots of research on creation of knowledge graphs, which has led to many domain knowledge graphs that have proven to be good in answering questions. Usually, these knowledge graphs contain object information (e.g. about food objects, recipes, people, books). To make such knowledge graphs *actionable*, it is important to link the contained **object knowledge** to **environment knowledge**. If robots shall use the knowledge graphs for action execution, they need to further include **action knowledge**.

ESWC 2024 Workshops and Tutorials Joint Proceedings, May 26-27, Heraklion, Greece

beetz@cs.uni-bremen.de (M. Beetz); cimiano@techfak.uni-bielefeld.de (P. Cimiano); michaela.kuempel@uni-bremen.de (M. Kümpel); enrico.motta@open.ac.uk (E. Motta); i.tiddi@vu.nl (I. Tiddi); jtoeberg@techfak.uni-bielefeld.de (J. Töberg)
0000-0002-7888-7444 (M. Beetz); 0000-0002-4771-441X (P. Cimiano); 0000-0002-0408-3953 (M. Kümpel); 0000-0003-0015-1952 (E. Motta); 0000-0001-7116-9338 (I. Tiddi); 0000-0003-0434-6781 (J. Töberg)

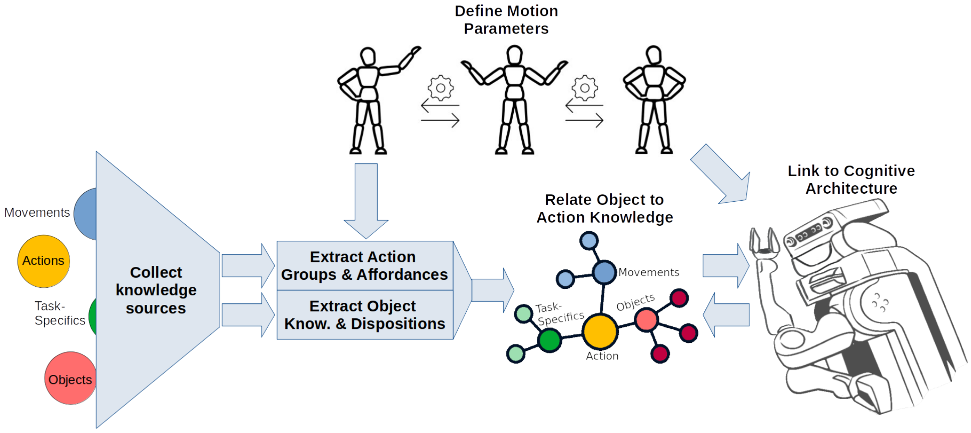
Define Motion Parameters
Movements
Actions
Task-Specifics
Objects
Collect knowledge sources
Extract Action Groups & Affordances
Extract Object Know. & Dispositions
Relate Object to Action Knowledge
Movements
Task-Specifics
Objects
Action
Link to Cognitive Architecture

Figure 1: The Knowledge Engineering Methodology proposed in [2] we use as the foundation for the tutorial.

This implies that actionable knowledge graphs do not aim at perfectly modeling object knowledge, but instead focus on reuse of existing knowledge sources and modeling and linking of environment and action knowledge in order for making the contained knowledge applicable in agent applications. This tutorial will detail the necessary concepts for creating an actionable knowledge graph for the example domain of *Cutting Fruits and Vegetables*, which shall be used by robotic agents to be able to infer the correct body motions for quartering an apple or dicing a cucumber.

2. Structure of the Tutorial

The tutorial is centered around the knowledge engineering methodology introduced in [2] and its application on the exemplary task of *Cutting Fruits & Vegetables*. In general, the methodology consists of five steps to create actionable knowledge graphs that a robot can employ to handle manipulation tasks, as can be examined in Figure 1. In the following we present a brief summarisation of these steps:

1) **Defining Motion Parameters**: Definition of the domain- and action-dependent parameters influencing the execution of the target manipulation action. An example is the knife position for cutting tasks.
2) **Collecting Knowledge Sources**: Collection of different sources for three types of knowledge: action knowledge, object knowledge & knowledge for linking action and object knowledge
3a) **Extraction of Action Groups & Affordances**: Collect information about the manipulation action and its associated synonyms and hyponyms. This information is used to organize different action verbs into groups based on similarities in their motion parameters. For each so called action group, a representative is chosen and their affordances are created.
3b) **Extraction of Object Knowledge & Dispositions**: Collect information about objects participating in the manipulation action (e.g. tools, environments, targets). Then collect information and concrete values for the task-specific object properties that influence the action execution. This knowledge is represented through dispositions.
4) **Relate Object to Action Knowledge**: Relate the action affordances to the object dispositions in an ontology by re-using relations from the SOMA [3] ontology.
5) **Link to Cognitive Architecture**: Map concepts in the generalized manipulation plan to their representation in the ontology and use the architecture's perception system to ground objects and their properties.

In this tutorial we present the whole methodology but focus on the steps 1), 3) and 4), which represent the knowledge collection and extraction from (Semantic) Web resources.

2.1. Defining Motion Parameters

In order to create an actionable knowledge graph for the domain of cutting fruits and vegetables, we first have to investigate motion parameters that influence action execution. For this, one can first investigate a lexical resource like WordNet [4] to find commonly used synonyms of cutting, such as slicing, dicing, or halving.

We then investigate how different action verbs influence task execution, which results in the following motion parameters:

- number of repetitions: Cutting tasks vary in the number of repetitions to be executed. Sometimes, a cut is only performed once, while other tasks require to cut the whole object.
- cutting position: Cutting tasks also vary in the applied cutting position. Halving requires a different position than slicing, for example.
- result object: Cutting tasks result in objects of different amount and shape.
- prior actions: Some objects require a prior action (such as peeling) to be executed.
- dependent tasks: Some tasks depend on prior tasks (i.e. quartering depends on halving).

2.2. Extraction of Relevant Action Knowledge from the Web

The relevant action knowledge we focus on consists of the different verbs that are associated with the manipulation action. This includes the main verb (e.g. cut) as well as all of its hyponyms and synonyms. Additionally, action knowledge covers the properties of the different verbs that distinguish their action execution and generally influence the manipulation action.

In the tutorial we showcase the action knowledge extraction for the exemplary task of *Cutting*. We begin by extracting all synonyms and hyponyms from WordNet [4] and VerbNet [5], two expertly created resources for lexical information and verb usage. For the verb *cut*, we extract 211 verbs from WordNet and 147 verbs from VerbNet. After pre-processing and duplicate removal, 181 verbs remain. These remaining verbs are then filtered based on their relevance for the domain using an instruction-focused corpus from WikiHow. We set a threshold of 100 occurrences in a specific part of an article across the whole corpus to warrant an inclusion of the verb in future steps. With this restriction, only 46 verbs remain. However, there is still a need for manual post-processing since some important verbs are missing (e.g. `halve` or `quarter`) or are very general and thus not relevant for cutting (e.g. `make` or `pull`).

Table 1

Comparison of different methods for extracting anatomical parts for a given fruit sorted based on their F1-score. In each column, we mark the three methods with the highest performance in **bold**.

Method	Acc.	Prec.	Rec.	Spec.	F1	Threshold
Recipe1M+ 2-Step	**.863**	**.824**	.636	**.948**	**.718**	Occ. in $\geq 1\%$ of steps
ChatGPT	.775	.556	**.909**	.724	**.690**	-
GPT-4	.700	.476	**.909**	.621	**.625**	-
CN Numberbatch	**.788**	.609	.636	.845	.622	Cossim ≥ 0.20
Recipe1M+ Bigrams	.688	.463	**.864**	.621	.603	Occ. in any step
Recipe1M+ 2-Step	.738	.517	.682	.759	.588	Occ. in $\geq 0.5\%$ of steps
Recipe1M+ Bigrams	.788	**.667**	.455	**.914**	.541	Occ. in $\geq 0.1\%$ of steps
CN Numberbatch	**.825**	**1.00**	.364	**1.00**	.533	Cossim ≥ 0.30
GloVe	.550	.348	.727	.483	.471	Cossim ≥ 0.25
GloVe	.688	.435	.455	.776	.444	Cossim ≥ 0.40
NASARI	.750	.571	.364	.897	.444	Cossim ≥ 0.75
GloVe	.738	.533	.364	.879	.432	Cossim ≥ 0.50
NASARI	.500	.295	.591	.466	.394	Cossim ≥ 0.50

2.3. Extraction of Relevant Object Knowledge from the Web

For the object knowledge, we focus on information about objects involved in the manipulation action, their properties, usage and their specific purpose. In general we showcase a similar pipeline to the one explained in Section 2.2. We begin by extracting all relevant objects from domain-specific taxonomies. For our focus on fruits and vegetables, we query the FoodOn [6] using SPARQL, resulting in 257 unique fruits and 31 unique vegetables. Since not all of these fruits and vegetables are equally relevant and we need enough information to exist to evaluate their task-specific properties, we again use instruction-focused corpora to filter them based on their occurrence data. In this case we also look at the recipe corpus Recipe1M+ [7] and only include fruits and vegetables that occur in 1% of any part of these two corpora. This filtering step results in 15 remaining fruits and one remaining vegetables.

Lastly, we present our ongoing efforts in automating the extraction of task-specific object property values. For this, we compare three different pre-trained embeddings (GloVe [8], NASARI [9] and ConceptNet Numberbatch [10]), two large language models (ChatGPT and GPT-4) as well as two techniques for extracting this information from the Recipe1M+ on the task of extracting the existing anatomical parts for a given fruit. Our preliminary results and their condition can be examined in Table 1.

2.4. Linking Action to Object Knowledge in the Ontology

For connecting and linking the action to the object knowledge, we rely on the concepts of *disposition* and *affordance*. In general, a *disposition* describes the property of an object, thereby enabling an agent to perform a certain task [11] as in *a knife can be used for cutting*, whereas an *affordance* describes what an object or the environment offers an agent [12] as in *an apple affords to be cut*.

In recent works like SOMA [3], both concepts are set in relation by stating that dispositions allow objects to participate in events realizing affordances, which are more abstract descriptions of dispositions. This is achieved in the TBOX by using the `affordsTask`, `affordsTrigger` and `hasDisposition` relations from SOMA. An example for the disposition of *Peelability* can be examined in Section 2.4.

```
hasDisposition some
    (Peelability
    and (affordsTask some Peeling)
    and (affordsTrigger only (classifies only Hand)))
```

Figure 2: Example for connecting an affordance (”Peeling with a hand”) to a disposition (”Peelability”) using relations from the SOMA ontology [3].

3. Tutorial Material

For the tutorial, we made our implementation available in Jupyter Notebooks found in a GitHub repository[1]. Participants are encouraged to download the notebooks and follow along, but the notebooks are presented in depth during the talks, so actual hands-on experience is optional.

Acknowledgments

The tutorial is organized by the SAIL Network in collaboration with the Joint Research Center on Cooperative and Cognition-enabled AI (CoAI JRC). The research towards this Tutorial has been partially supported by the German Federal Ministy of Education and Research; Project-ID 16DHBKI047 “IntEL4CoRo - Integrated Learning Environment for Cognitive Robotics”, University of Bremen as well as the German Research Foundation DFG, as part of CRC (SFB) 1320 “EASE - Everyday Activity Science and Engineering”, University of Bremen (http://www.ease-crc.org/). The research was conducted in subproject R04 “Cognition-enabled execution of everyday actions”.

[1] https://github.com/Food-Ninja/Tutorial_ESWC_HHAI

References

[1] M. Kümpel, Actionable knowledge graphs - how daily activity applications can benefit from embodied web knowledge, 2024. doi:10.26092/elib/2936.

[2] M. Kümpel, J.-P. Töberg, V. Hassouna, P. Cimiano, M. Beetz, Towards a Knowledge Engineering Methodology for Flexible Robot Manipulation in Everyday Tasks, in: International Workshop on Actionable Knowledge Representation and Reasoning for Robots (AKR^3), Heraklion, Crete, Greece, 2024.

[3] D. Beßler, R. Porzel, M. Pomarlan, A. Vyas, S. Höffner, M. Beetz, R. Malaka, J. Bateman, Foundations of the Socio-physical Model of Activities (SOMA) for Autonomous Robotic Agents, in: Formal Ontology in Information Systems, volume 344 of *Frontiers in Artificial Intelligence and Applications*, IOS Press, Amsterdam, 2022, pp. 159–174. URL: https://ebooks.iospress.nl/doi/10.3233/FAIA210379. arXiv:2011.11972.

[4] G. A. Miller, WordNet: A Lexical Database for English, Communications of the ACM 38 (1995) 39–41. doi:10.1145/219717.219748.

[5] K. K. Schuler, VerbNet: A Broad-Coverage, Comprehensive Verb Lexicon, Ph.D. thesis, University of Pennsylvania, 2005.

[6] D. M. Dooley, E. J. Griffiths, G. S. Gosal, P. L. Buttigieg, R. Hoehndorf, M. C. Lange, L. M. Schriml, F. S. L. Brinkman, W. W. L. Hsiao, FoodOn: A harmonized food ontology to increase global food traceability, quality control and data integration, npj Sci Food 2 (2018) 23. doi:10.1038/s41538-018-0032-6.

[7] J. Marín, A. Biswas, F. Ofli, N. Hynes, A. Salvador, Y. Aytar, I. Weber, A. Torralba, Recipe1M+: A Dataset for Learning Cross-Modal Embeddings for Cooking Recipes and Food Images, IEEE Transactions on Pattern Analysis and Machine Intelligence 43 (2021) 187–203. doi:10.1109/TPAMI.2019.2927476.

[8] J. Pennington, R. Socher, C. Manning, Glove: Global Vectors for Word Representation, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Doha, Qatar, 2014, pp. 1532–1543. doi:10.3115/v1/D14-1162.

[9] J. Camacho-Collados, M. T. Pilehvar, R. Navigli, NASARI: A Novel Approach to a Semantically-Aware Representation of Items, in: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, CO, 2015, pp. 567–577. URL: http://aclweb.org/anthology/N/N15/N15-1059.pdf.

[10] R. Speer, J. Chin, C. Havasi, ConceptNet 5.5: An Open Multilingual Graph of General Knowledge, AAAI 31 (2017). doi:10.1609/aaai.v31i1.11164.

[11] M. T. Turvey, Ecological foundations of cognition: Invariants of perception and action., in: H. L. Pick, P. W. van den Broek, D. C. Knill (Eds.), Cognition: Conceptual and Methodological Issues., American Psychological Association, Washington, 1992, pp. 85–117. doi:10.1037/10564-004.

[12] M. H. Bornstein, J. J. Gibson, The Ecological Approach to Visual Perception, The Journal of Aesthetics and Art Criticism 39 (1980) 203. doi:10.2307/429816. arXiv:10.2307/429816.