

Tutorial on Contextualizing and Executing Robot Manipulation Plans Using Web Knowledge

Michael Beetz¹, Philipp Cimiano², Michaela Kümpel¹, Enrico Motta³, Ilaria Tiddi⁴
& Jan-Philipp Töberg²

¹Institute for Artificial Intelligence, University of Bremen, Bremen, Germany

²Cluster of Excellence Cognitive Interaction Technology (CITEC), Bielefeld University, Bielefeld, Germany

³Knowledge Media Institute, The Open University, Milton Keynes, United Kingdom

⁴Vrije Universiteit Amsterdam, The Netherlands

3rd International Conference on Hybrid Human-Artificial Intelligence
June 10th, 2024



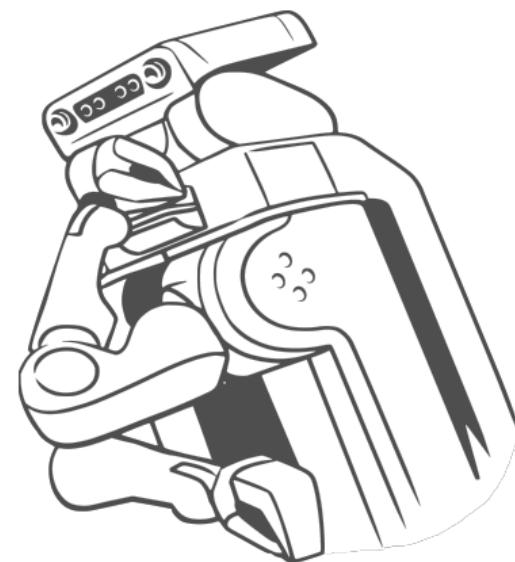
Before we start...

Please download the material from GitHub:

<https://github.com/Food-Ninja/Tut-ExeWebKnow>

Outline

- 1 Motivation
- 2 Knowledge Engineering Methodology
- 3 Extraction of Relevant Action and Object Knowledge from the Web
- 4 Knowledge Engineering Methodology
- 5 Accessing Web Knowledge & Knowledge Graphs for Action Parameterisation



Robots in Open World Situations

State of the Art

- We see many household robots (lawn mowers, vacuum cleaners)
- Industry (car manufacturers etc.) is relying on robots
- Many people have envisioned meal preparation robots
→ but they are yet to be seen in stores

Robots in Open World Situations

Robots perform well ...

- ... in static environments (car manufacturer)
- ... given static (or known) objects
- ... given known tasks

Robots in Open World Situations

Robots perform *well* ...

- ... in static environments (car manufacturer)
- ... given static (or known) objects
- ... given known tasks

Problem

Households are *open worlds* ⇒ unknown environments containing unknown objects and necessitating unknown actions.

Households ...

- ... are never static
- ... have different objects at different locations
- ... are open to unknown tasks

Robots in Open World Situations

Solutions

- Acquiring knowledge via human-robot interaction (learning from demonstration, imitation learning)
- **Dynamically building a knowledge graph / ontology to assist a task planner**
 - usually used to infer the next step to perform in an action
- Use generative Large Language Models (LLMs) to assist a task planner (e.g. *SayCan* [Ahn+22] or *Code As Policies* [Lia+23])
- Translating language to motion (works well for simple motions)

Robots in Open World Situations

Solutions

- Acquiring knowledge via human-robot interaction (learning from demonstration, imitation learning)
- **Dynamically building a knowledge graph / ontology to assist a task planner**
 - usually used to infer the next step to perform in an action
- Use generative Large Language Models (LLMs) to assist a task planner (e.g. *SayCan* [Ahn+22] or *Code As Policies* [Lia+23])
- Translating language to motion (works well for simple motions)

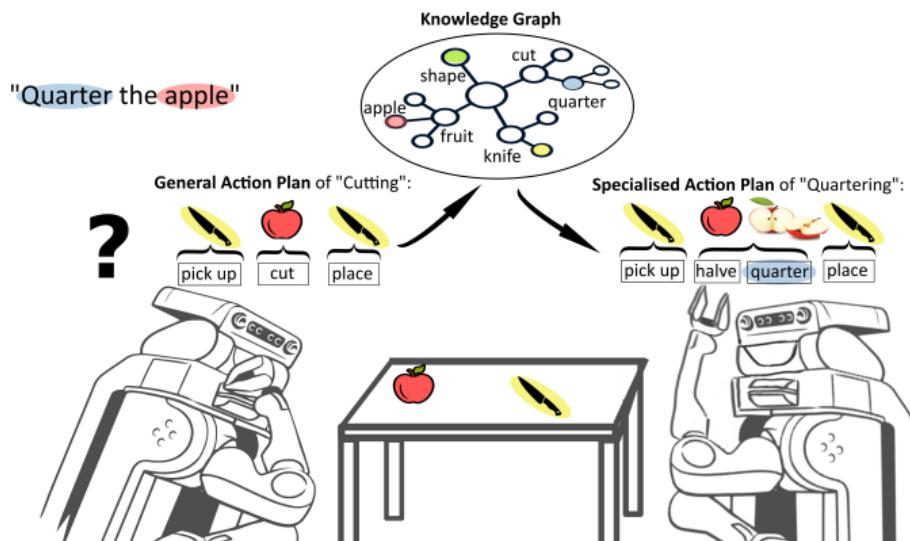
For the domain of cutting fruits and vegetables:

- learning optimal cutting (good when the goal is to have one optimal cut)
- learning number of slices to cut

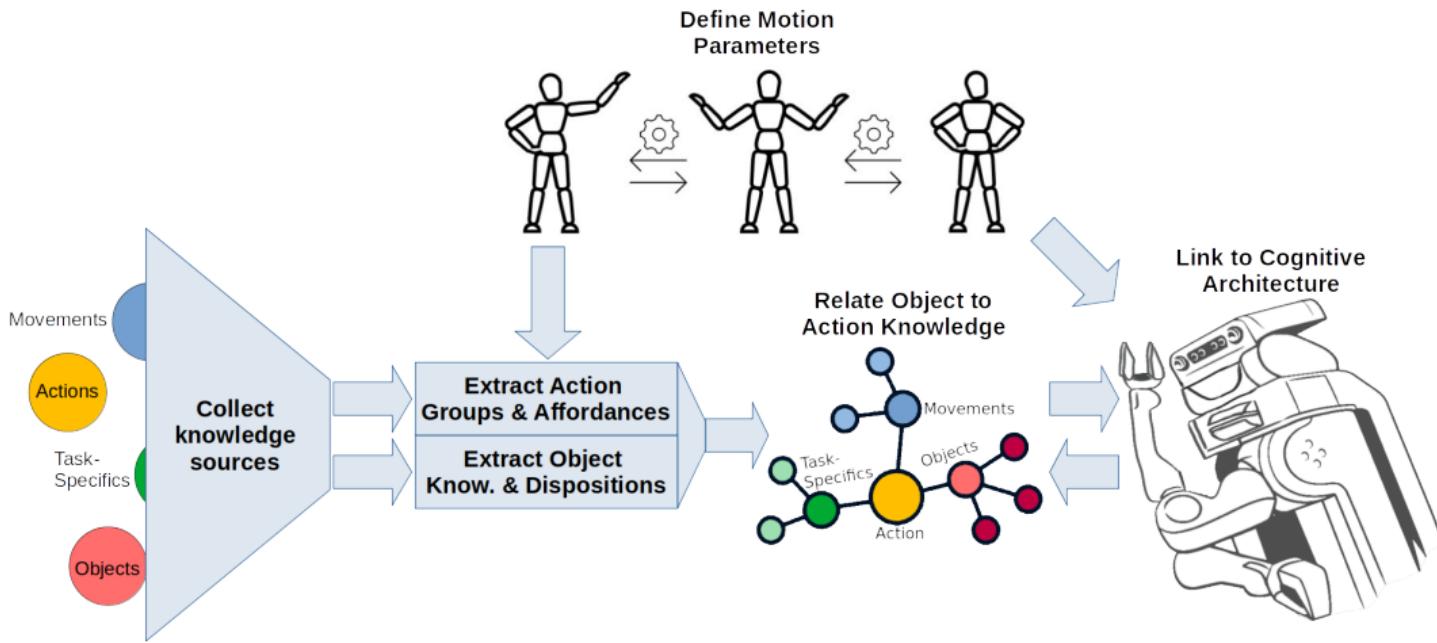
Robots in Open World Situations

Our idea [Küm+24a; Küm+24b]

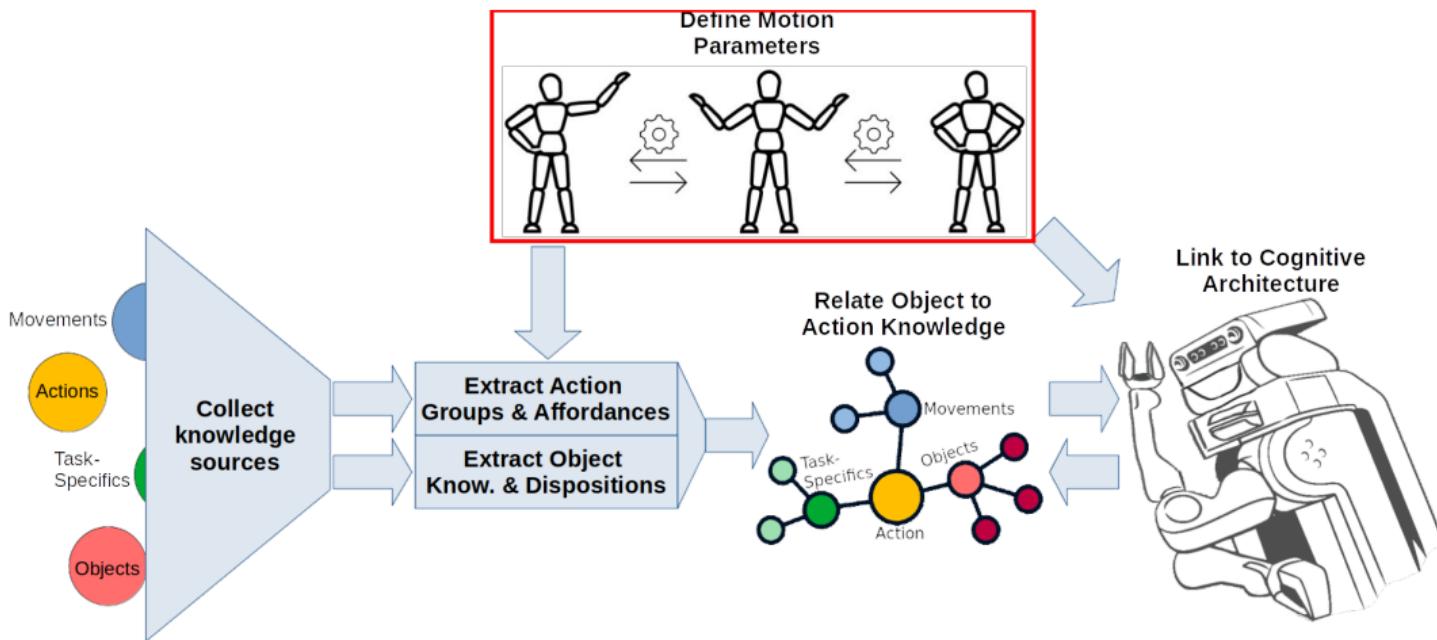
Given a robot that knows how to cut a slice of bread, enable it to quarter an apple, or dice a cucumber



Knowledge Engineering Methodology [Küm+24b]



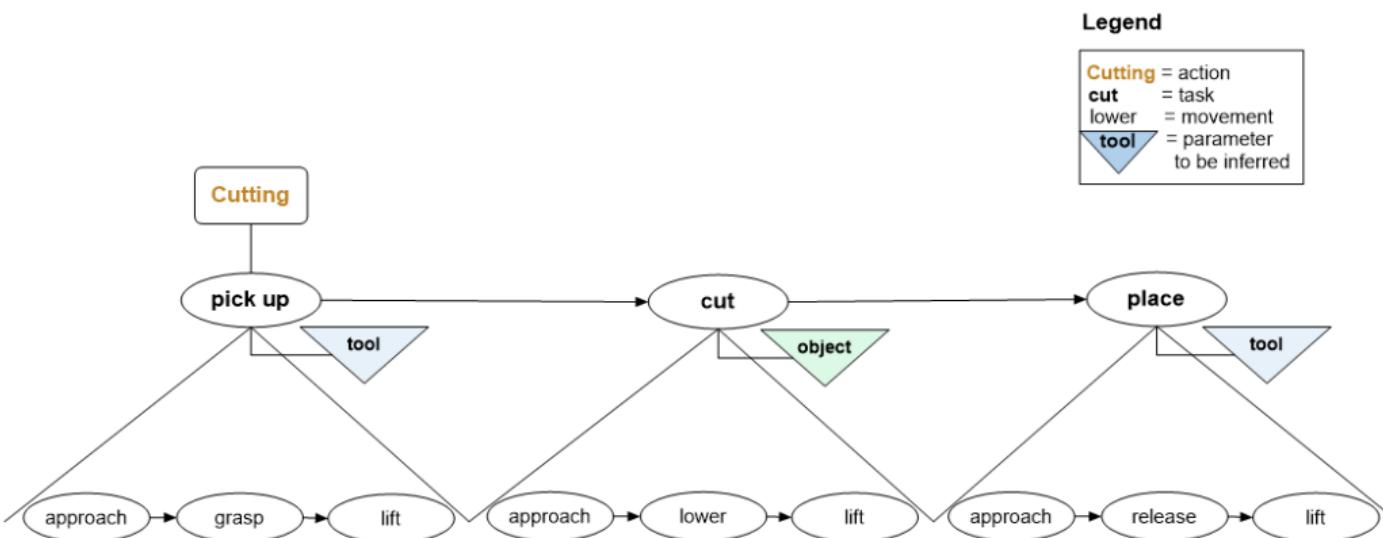
Knowledge Engineering Methodology: Step 1



Knowledge Engineering Methodology: Step 1

General Action Plans

To enable a robot to perform task variations, we have to look at action plans

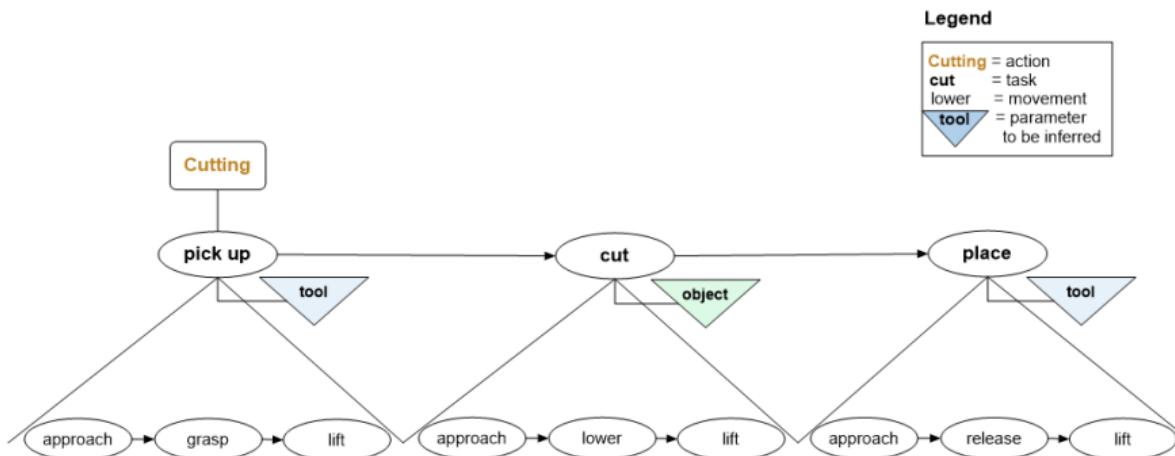


Knowledge Engineering Methodology: Step 1

General Action Plans

With the given plan

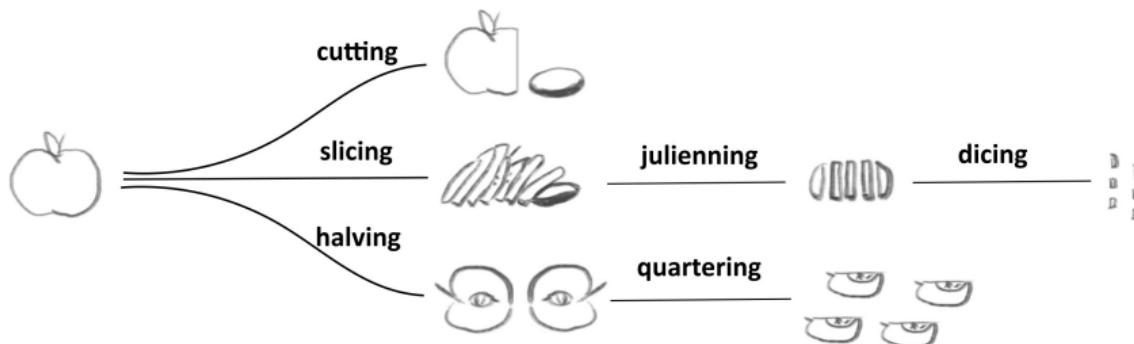
- a robot can cut a given object once at a set position
- How can a robot be enabled to perform different cutting tasks?



Knowledge Engineering Methodology: Step 1

Generate Action Groups

For a general action, inspect task variations and how they influence body movements

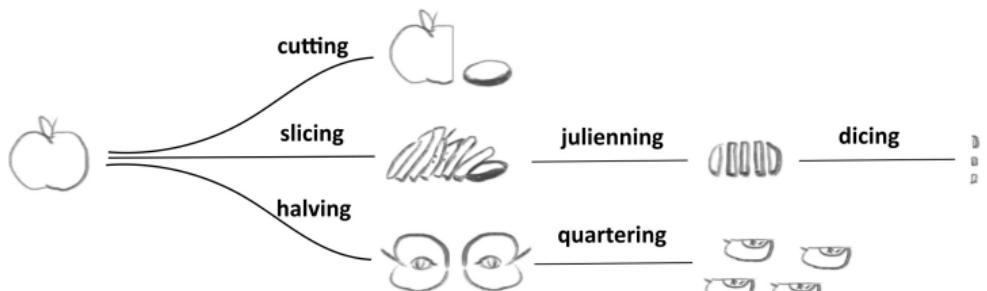


Knowledge Engineering Methodology: Step 1

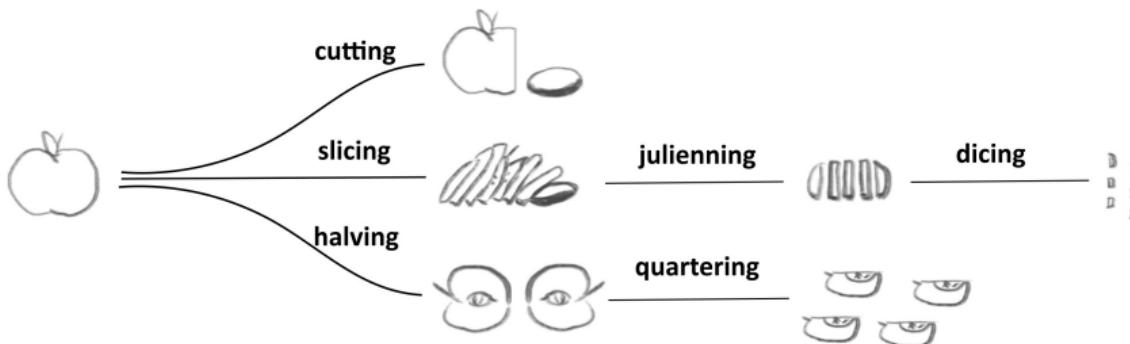
Generate Action Groups

Cutting tasks differ by

- the cutting position (middle or end of an object)
- number of repetitions
- object being cut (important for perception)
- resulting objects (number, shape)



Knowledge Engineering Methodology: Step 1



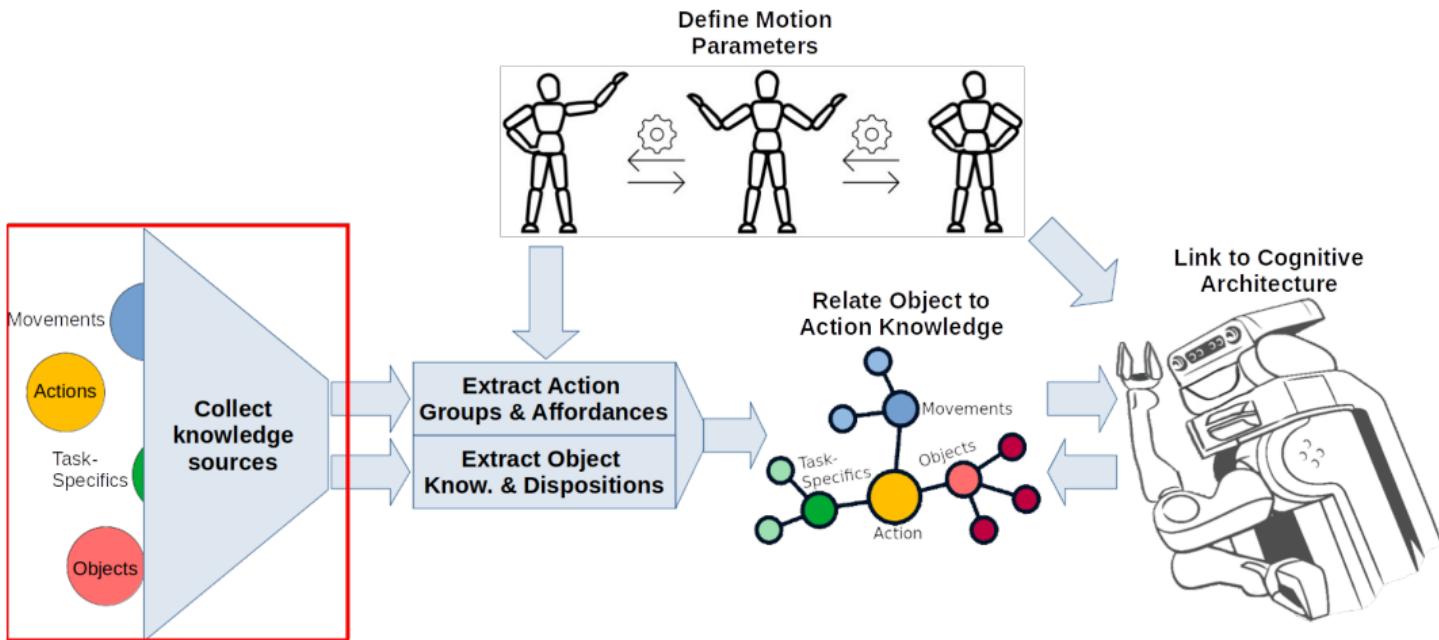
Action Groups

Hypothesis:

Different action verbs can be classified into these action groups since they achieve similar results.

Example: dicing, cubing, chopping all result in cube-shaped pieces

Knowledge Engineering Methodology: Step 2



Knowledge Engineering Methodology: Step 2

Action Knowledge

- Action verbs that are associated with the manipulation action
- Properties of the action verbs necessary for manipulation
- Information for grouping of similar actions into *action groups*

Knowledge Engineering Methodology: Step 2

Action Knowledge

- Action verbs that are associated with the manipulation action
- Properties of the action verbs necessary for manipulation
- Information for grouping of similar actions into *action groups*

Possible Sources

- Action Verbs: VerbNet [Sch05], FrameNet [BFL98], WordNet [Mil95]
- In the domain context (meal preparation)
- Differentiating between preparation tasks (peeling, filleting, coring) and cutting tasks (slicing, snipping, cubing)

Knowledge Engineering Methodology: Step 2

Object Knowledge

- Relevant information about involved objects, their usage & their purpose
- Properties for grounding basic manipulation actions (e.g. grasp, hold)
- Task-specific object knowledge

Knowledge Engineering Methodology: Step 2

Object Knowledge

- Relevant information about involved objects, their usage & their purpose
- Properties for grounding basic manipulation actions (e.g. grasp, hold)
- Task-specific object knowledge

Possible Sources

- Food taxonomy: FoodOn food ontology [Doo+18]
- Food classification: plant ontology [Jai+05]
- Food parts: LLMs, biology textbooks (e.g. [CLW18])

Knowledge Engineering Methodology: Step 2

Task-Specific Knowledge

- Relevant information about involved object parts
- Information about tools
- Information about environment (e.g. object orientation)

Knowledge Engineering Methodology: Step 2

Task-Specific Knowledge

- Relevant information about involved object parts
- Information about tools
- Information about environment (e.g. object orientation)

Possible Sources

- Object parts and their involvement in meal preparation: WikiHow, Recipe1M+ [Mar+21]
- Tools for action execution: Product websites

Knowledge Engineering Methodology: Step 2

Align everything using top-level ontologies

Reuse a top-level ontology, e.g.

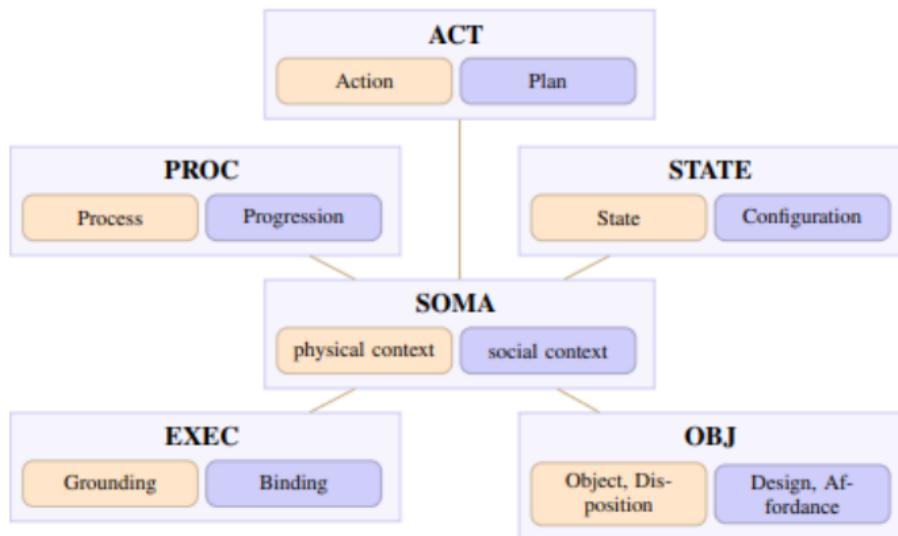
- Schema.org
- basic formal ontology bfo [ASS15]:
https://ncorwiki.buffalo.edu/index.php/BFO_Release_History
- DOLCE+DnS Ultralite (DUL) [PG16]:
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl>

→ only DUL relates agents to objects and actions they are involved in

Knowledge Engineering Methodology: Step 2

SOMA (Socio-Physical Model of Activities) [Beß+22]

Reuse an upper level ontology: SOMA, which extends DUL with the focus on household activities

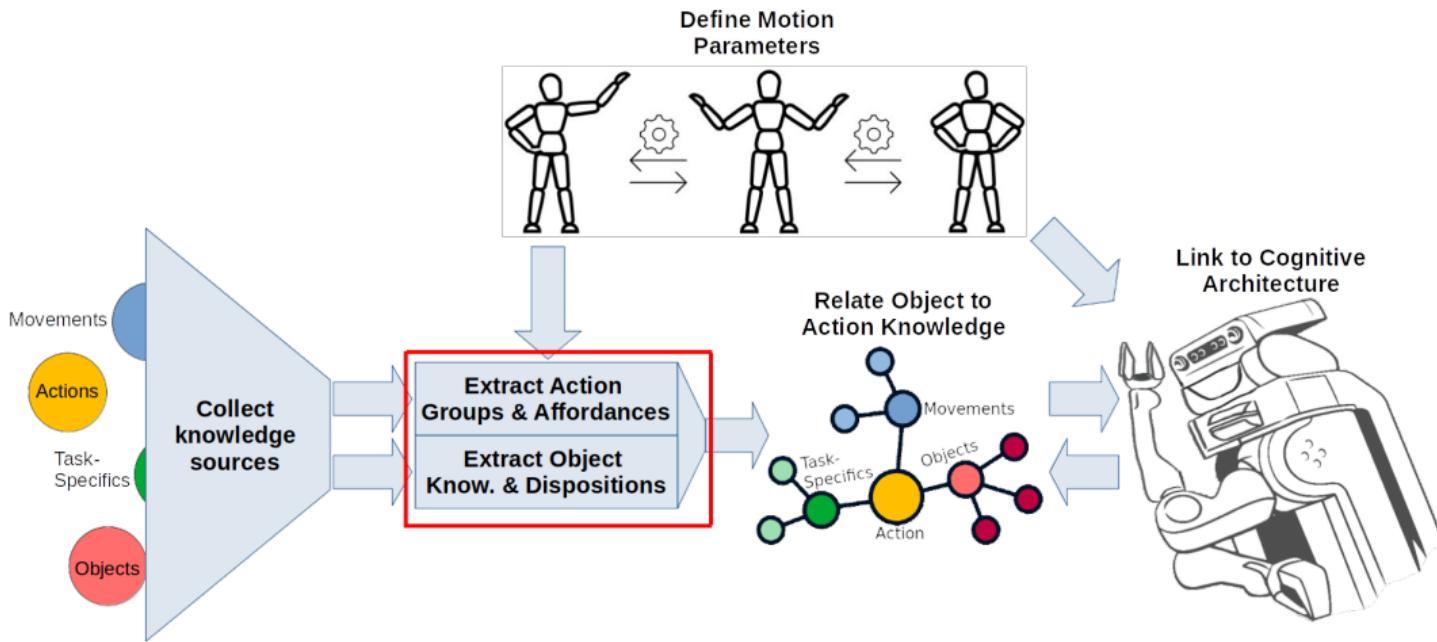


Knowledge Engineering Methodology: Step 2

Demonstration 01

Jupyter Notebook on Ontology Alignment

Knowledge Engineering Methodology: Step 3



Knowledge Engineering Methodology: Step 3

Affordance [BG80]

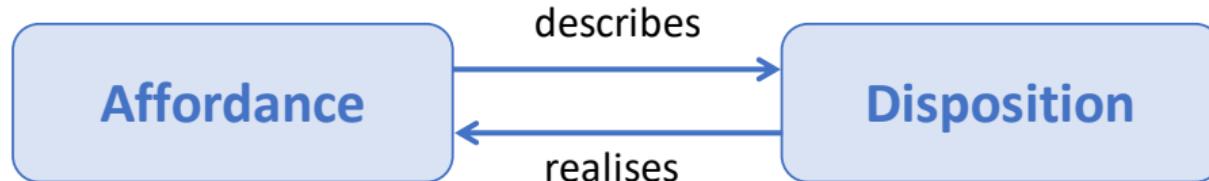
Describes what an object or the environment offers an agent.

Example: *An apple affords to be cut*

Disposition [Tur92]

Describes a property of an object that enables an agent to perform a certain task.

Example: *A knife can be used for cutting*



Knowledge Engineering Methodology: Step 3a

Extract Action Knowledge:

- 1 Collect synonyms & hyponyms of the manipulation action
- 2 Filter the collected verbs using instruction-focused corpora (e.g. WikiHow, Recipe1M+ [Mar+21])
- 3 Analyse the motion patterns associated with each remaining verb
- 4 Create *action groups* based on the overlaps in motion parameters
- 5 Create affordances for each group representative

Knowledge Engineering Methodology: Step 3b

Extract Object Knowledge:

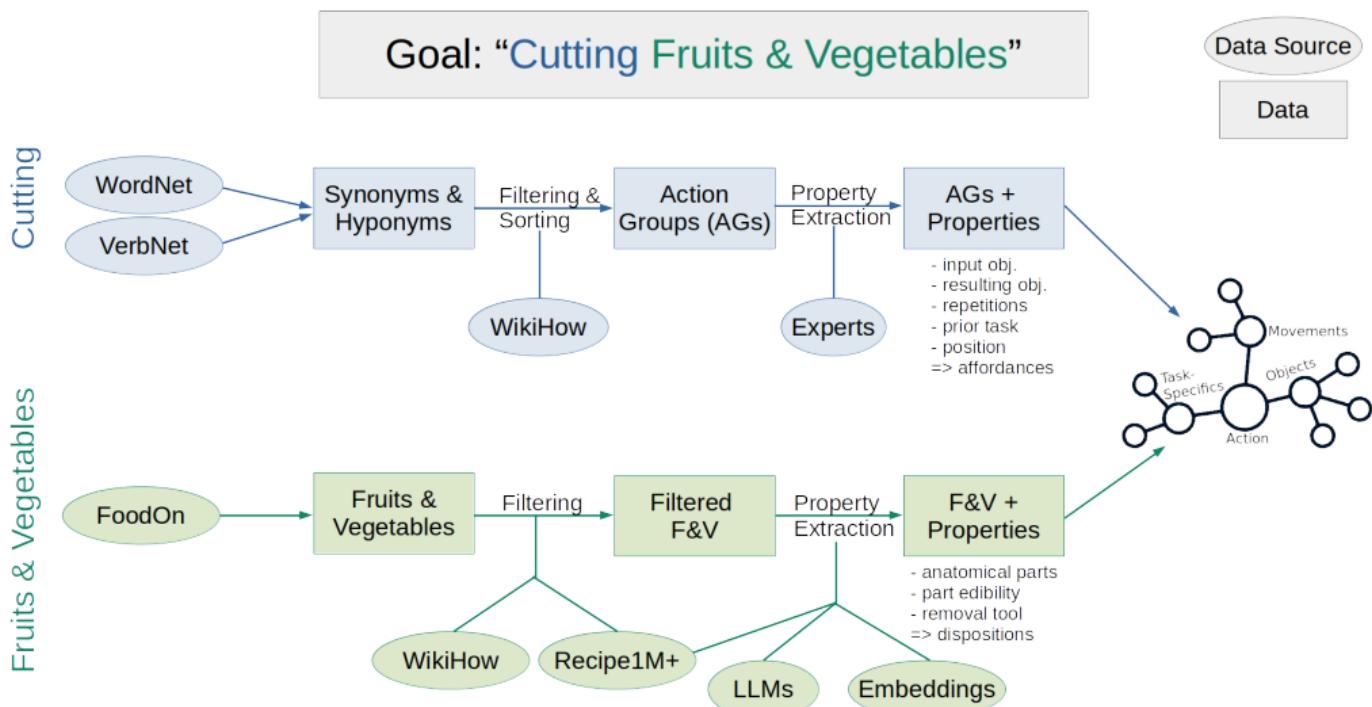
- 1 Collect objects from domain-specific object taxonomies (e.g. FoodOn [Doo+18], PlantOntology [Jai+05])
- 2 Filter the collected objects using instruction-focused corpora (e.g. WikiHow, Recipe1M+ [Mar+21])
- 3 Choose the task-specific object properties
- 4 Extracting property values for the remaining objects
- 5 Create dispositions based on these properties

Knowledge Engineering Methodology: Step 3a (Action Groups)

TABLE I: Analysing the occurrences of 15 different hyponyms for *cut* in the WikiHow data from [10].

	Verb	Occurrences	Action Group/ Cutting Position	Goal/ Repetitions	Input Object/ Prior Task
	cut pare carve saw	23486 921 888 149	cutting/ slicing pos	one slice/ = 1 repetitions	food/ no prior task
	slice snip sliver	8200 162 54	slicing/ slicing pos	multiple slices/ > 1 repetitions	food or food part/ no prior task
	julienné	189	juliennning/ slicing pos	multiple stripes/ > 1 repetitions	food slices / slicing
	chop mince dice cube	9221 2164 1631 516	dicing/ slicing pos	multiple cubes/ > 1 repetitions	food stripe/ juliennning
	halve	346	halving/ halving pos	halves/ = 1 repetitions	food/ no prior task
	quarter	125	quartering/ halving pos	quarters/ = 1 repetitions	food halve/ halving

Extraction of Action and Object Knowledge

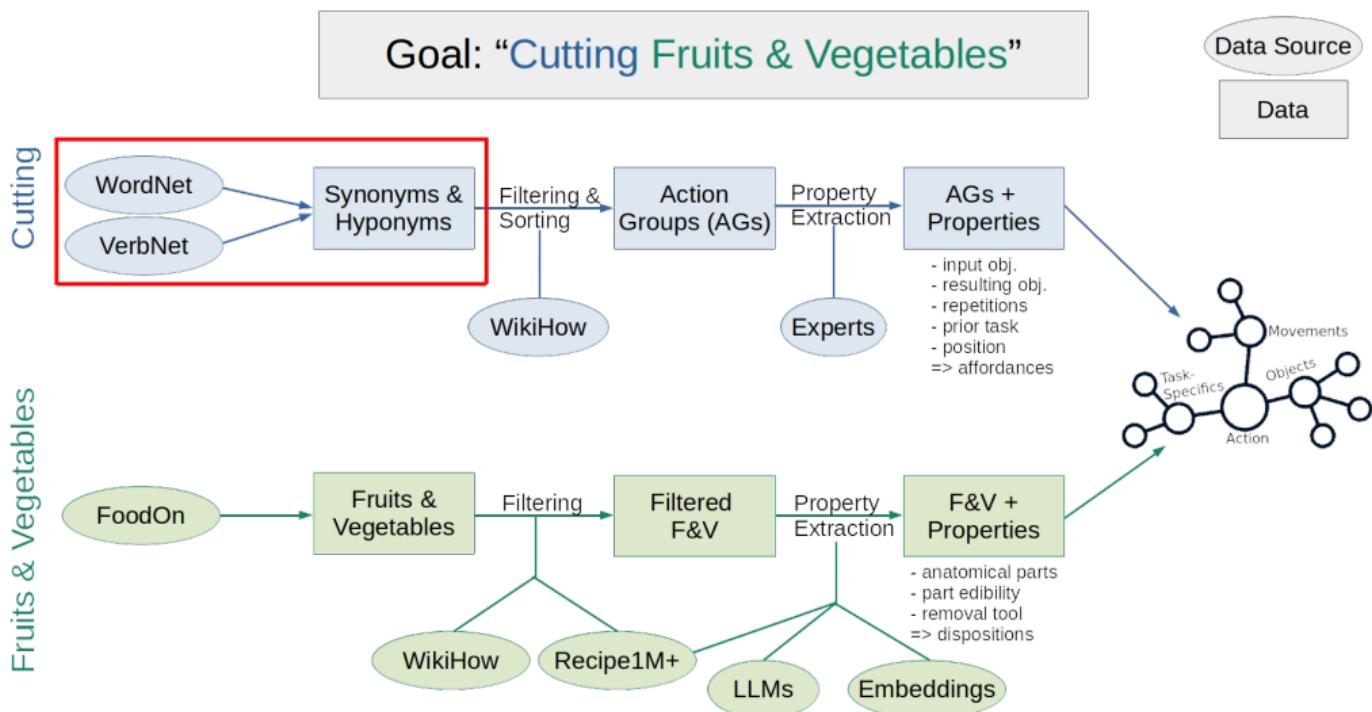


Extraction of Action Knowledge

Demonstration 02

Jupyter Notebook on Action Knowledge Extraction:
`Tut-ExeWebKnow/Knowledge Extraction/action_extract.ipynb`

Extraction of Action Knowledge



Synonyms and Hyponyms - Definition

Synonyms

- Relation between two words
- Different words with the same meaning
- Example: *movie* is a synonym for *film*

Synonyms and Hyponyms - Definition

Synonyms

- Relation between two words
- Different words with the same meaning
- Example: *movie* is a synonym for *film*

Hyponyms

- Relation between two words
- One word is a more specific / concrete definition of the other
- Example: *apple* is a hyponym to *fruit*

Resources for Synonyms and Hyponyms

WordNet [Mil95]

- Machine-readable lexical database containing (English) nouns, verbs, adjectives & adverbs
- Words are organized into *sets of synonyms* (= Synsets)
- Contains $\sim 166,000$ synsets

Resources for Synonyms and Hyponyms

WordNet [Mil95]

- Machine-readable lexical database containing (English) nouns, verbs, adjectives & adverbs
- Words are organized into *sets of synonyms* (= Synsets)
- Contains $\sim 166,000$ synsets

VerbNet [Sch05]

- Hierarchical, domain-independent verb lexicon for the English language
- Verbs are organized in classes that describe their thematic roles (e.g. Agent, Destination or Product), syntactic frames and semantic predicate structures
- Contains $\sim 5,300$ different word senses

Cut in WordNet [Mil95]

Verb

- S: (V) cut (separate with or as if with an instrument)
- S: (V) reduce, cut down, cut back, trim, trim down, trim back, cut, bring down (cut down on; make a reduction in)
- S: (V) swerve, sheer, curve, trend, veer, slue, slew, cut (turn sharply; change direction abruptly)
- S: (V) cut (make an incision or separation)
- S: (V) cut (discharge from a group)
- S: (V) cut (form by probing, penetrating, or digging)
- S: (V) cut, tailor (style and tailor in a certain fashion)
- S: (V) cut (hit (a ball) with a spin so that it turns in the opposite direction)
- S: (V) write out, issue, make out, cut (make out and issue)
- S: (V) edit, cut, edit out (cut and assemble the components of)
- S: (V) cut, skip (intentionally fail to attend)
- S: (V) hack, cut (be able to manage or manage successfully)
- S: (V) cut (give the appearance or impression of)
- S: (V) cut (move (one's fist))
- S: (V) cut (pass directly and often in haste)
- S: (V) cut (pass through or across)
- S: (V) cut (make an abrupt change of image or sound)
- S: (V) cut (stop filming)
- S: (V) cut (make a recording of)
- S: (V) cut (record a performance on (a medium))
- S: (V) cut, burn (create by duplicating data)
- S: (V) cut (form or shape by cutting or incising)
- S: (V) cut (perform or carry out)
- S: (V) cut (function as a cutting instrument)
- S: (V) cut (allow inclusion or separation)
- S: (V) cut (divide a deck of cards at random into two parts to make selection difficult)
- S: (V) switch off, cut, turn off, turn out (cause to stop operating by disengaging a switch)
- S: (V) cut (reap or harvest)
- S: (V) cut (fell by sawing; hew)
- S: (V) cut (penetrate injuriously)
- S: (V) ignore, disregard, snub, cut (refuse to acknowledge)
- S: (V) cut (shorten as if by severing the edges or ends of)
- S: (V) cut, prune, rationalize, rationalise (weed out unwanted or unnecessary things)
- S: (V) cut (dissolve by breaking down the fat of)
- S: (V) cut (have a reducing effect)
- S: (V) cut, cut off (cease, stop)
- S: (V) abridge, foreshorten, abbreviate, shorten, cut, contract, reduce (reduce in scope while retaining essential elements)
- S: (V) dilute, thin, thin out, reduce, cut (lessen the strength or flavor of a solution or mixture)
- S: (V) cut (have grow through the gums)
- S: (V) cut (grow through the gums)
- S: (V) geld, cut (cut off the testicles (of male animals such as horses))

<http://wordnetweb.princeton.edu/perl/webwn?s=cut>

33/90

Cut in VerbNet [Sch05]

CUT-21.1-1

Parent Classes:
CUT-21.1

This is not the parent class. For full class view click below
[Full Class View](#)

cut-21.1
cut-21.1-1

Member Verb Lemmas-topfile

CHIP CHOP CLIP CUT HACK HEW RIP SAW SCARIFY
SCRAPE SCRATCH SLASH SLICE SNIP REAM

ROLES:
Patient [+body_part | +ref!]

EXAMPLE:
Carol cut her finger on the sharp edge.
[SHOW DEPENDENCY PARSE TREE](#)

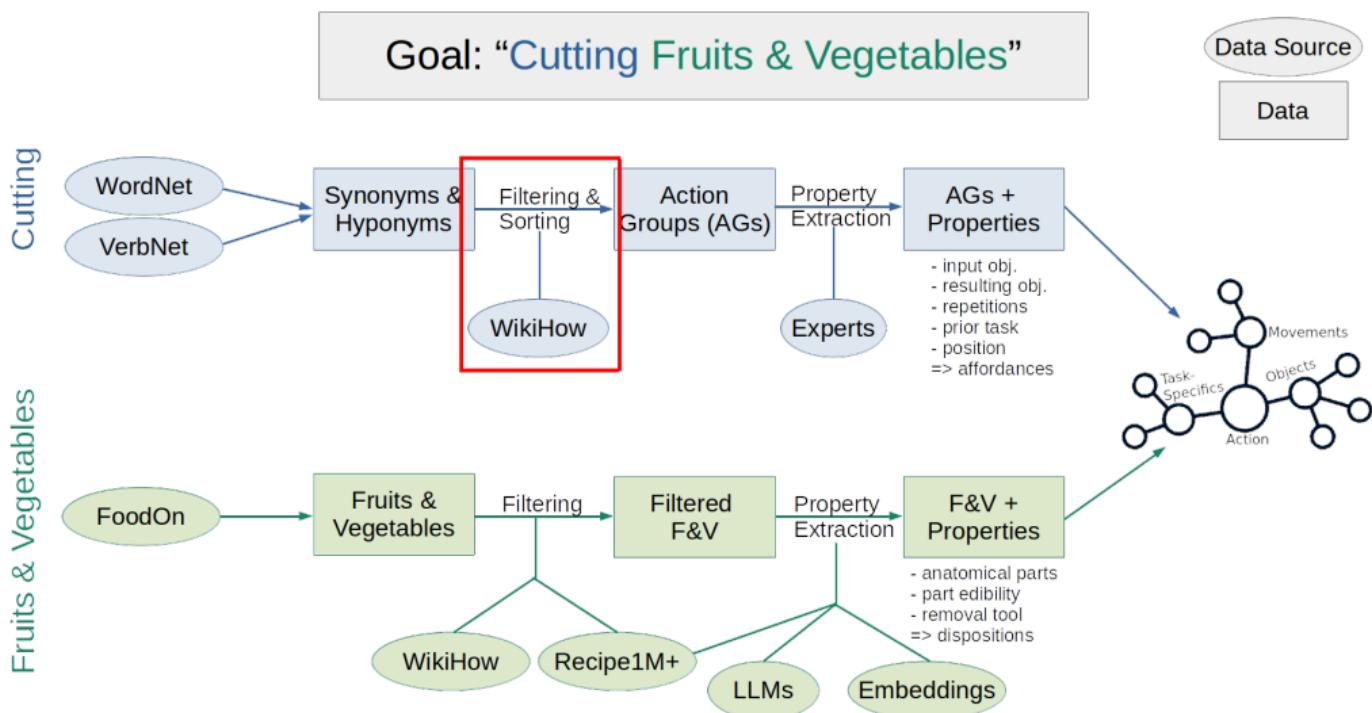
SYNTAX:
Agent VERB Patient { with } Instrument

SEMANTICS:

~ DEGRADATION_MATERIAL_INTEGRITY(e1 , Patient)
DO(e2 , Agent)
UTILIZE(e2 , Agent , Instrument)
MANNER(e2 , V_Movement , Agent)
CONTACT(e2 , Patient , Instrument)
DEGRADATION_MATERIAL_INTEGRITY(e3 , Patient)
CAUSE(e2 , e3)

<https://uvi.colorado.edu/verbnet/cut-21.1>

Extraction of Action Knowledge



Filtering the Extracted Verbs

- 1 Pre-Processing to split all synonyms/hyponyms that consist of 2 words (e.g. `snip_off` ⇒ include only `snip`)
- 2 Remove all duplicates
- 3 Use the *WikiHowAnalysisTool* [Töb22] to count the occurrences of the remaining verbs in different parts of a WikiHow article
- 4 Remove all words that do not occur more than a certain number of times

WikiHow Article Structure

ALCOHOLIC DRINKS » WINE

How to Pour a Glass of Champagne

Title

Co-authored by [wikiHow Staff](#) and 19 contributors

Last Updated: May 6, 2021 References Tested

Champagne is a bubbly wine made in France. This celebratory drink is very delicate, and requires unique procedures in proper storing, opening, and pouring to preserve its flavor and bubbles. Before pouring, you'll need to make sure it's at an ideal temperature, not too warm or too cold. When opening the bottle, hold it properly and avoid shaking it. When pouring, hold the glass at an angle and only fill it halfway so that you and your guests enjoy the champagne's unique aroma and taste.

[Download Article](#)

PARTS

- [Maintaining the Ideal Temperature](#)
- [Opening the Bottle](#)
- [Pouring the Champagne](#)

OTHER SECTIONS

- [Questions & Answers](#)
- [Video](#)
- [Tips and Warnings](#)
- [Related Articles](#)
- [References](#)

Title Description

Step Headline

Part 3 Pouring the Champagne Method [Download Article](#)



Step Image

4 Tip the bottle and slowly pour the champagne down the side of the glass. With your glass at a 45-degree angle, carefully tip the champagne bottle and begin pouring. Make sure you aim the liquid so that it touches the side of the glass as it pours out. This prevents the build-up of foam and loss of bubbles.^[1]

- Do not let the bottle touch the glass. Champagne is sometimes stored in dusty areas, so do not risk getting your glass dirty.

Step Description

WikiHow Article Structure

ALCOHOLIC DRINKS » WINE

How to Pour a Glass of Champagne

Title

Co-authored by [wikiHow Staff](#) and 19 contributors

Last Updated: May 6, 2021 References Tested

Champagne is a bubbly wine made in France. This celebratory drink is very delicate, and requires unique procedures in proper storing, opening, and pouring to preserve its flavor and bubbles. Before pouring, you'll need to make sure it's at an ideal temperature, not too warm or too cold. When opening the bottle, hold it properly and avoid shaking it. When pouring, hold the glass at an angle and only fill it halfway so that you and your guests enjoy the champagne's unique aroma and taste.

[Download Article](#)

PARTS

- [Maintaining the Ideal Temperature](#)
- [Opening the Bottle](#)
- [Pouring the Champagne](#)

OTHER SECTIONS

- [Questions & Answers](#)
- [Video](#)
- [Tips and Warnings](#)
- [Related Articles](#)
- [References](#)

Title Description

Step Headline

Part 3 Pouring the Champagne Method [Download Article](#)



Step Image

4 Tip the bottle and slowly pour the champagne down the side of the glass. With your glass at a 45-degree angle, carefully tip the champagne bottle and begin pouring. Make sure you aim the liquid so that it touches the side of the glass as it pours out. This prevents the build-up of foam and loss of bubbles.^[1]

- Do not let the bottle touch the glass. Champagne is sometimes stored in dusty areas, so do not risk getting your glass dirty.

Step Description

⇒ Focus on step description since it provides the highest amount of information

Extraction of Action Knowledge - Results I

- 1 Synonyms/Hyponyms from WordNet: 211
Synonyms/Hyponyms from VerbNet: 147
- 2 Duplicate removal: 181
- 3 No. of verbs occurring in at least 1 article: 131
- 4 No. of verbs with ≥ 100 occurrences: 46

Extraction of Action Knowledge - Results II

46 Remaining verbs:

bite, break, bring, bruise, build, burn, **carve**, check, **chop**, condense, **cube**, curl, **cut**,
dice, dilute, draw, drop, kill, make, **pare**, peel, play, powder, pull, push, reduce, rinse,
roll, **scrape**, set, shape, shave, skip, **slice**, slip, spill, **split**, strain, switch, take, tap,
tear, **trim**, turn, write

Extraction of Action Knowledge - Results II

46 Remaining verbs:

bite, break, bring, bruise, build, burn, **carve**, check, **chop**, condense, **cube**, curl, **cut**,
dice, dilute, draw, drop, kill, make, **pare**, peel, play, powder, pull, push, reduce, rinse,
roll, **scrape**, set, shape, shave, skip, **slice**, slip, spill, **split**, strain, switch, take, tap,
tear, **trim**, turn, write

Findings & Challenges:

- Only a fraction of verbs are really relevant for **cutting fruits & vegetables**
- Some verbs are included due to their generality (e.g. make or pull)
- Some verbs should not be performed by a robot (e.g. bite or kill)
- Relevant verbs are missing (e.g. halve or quarter)

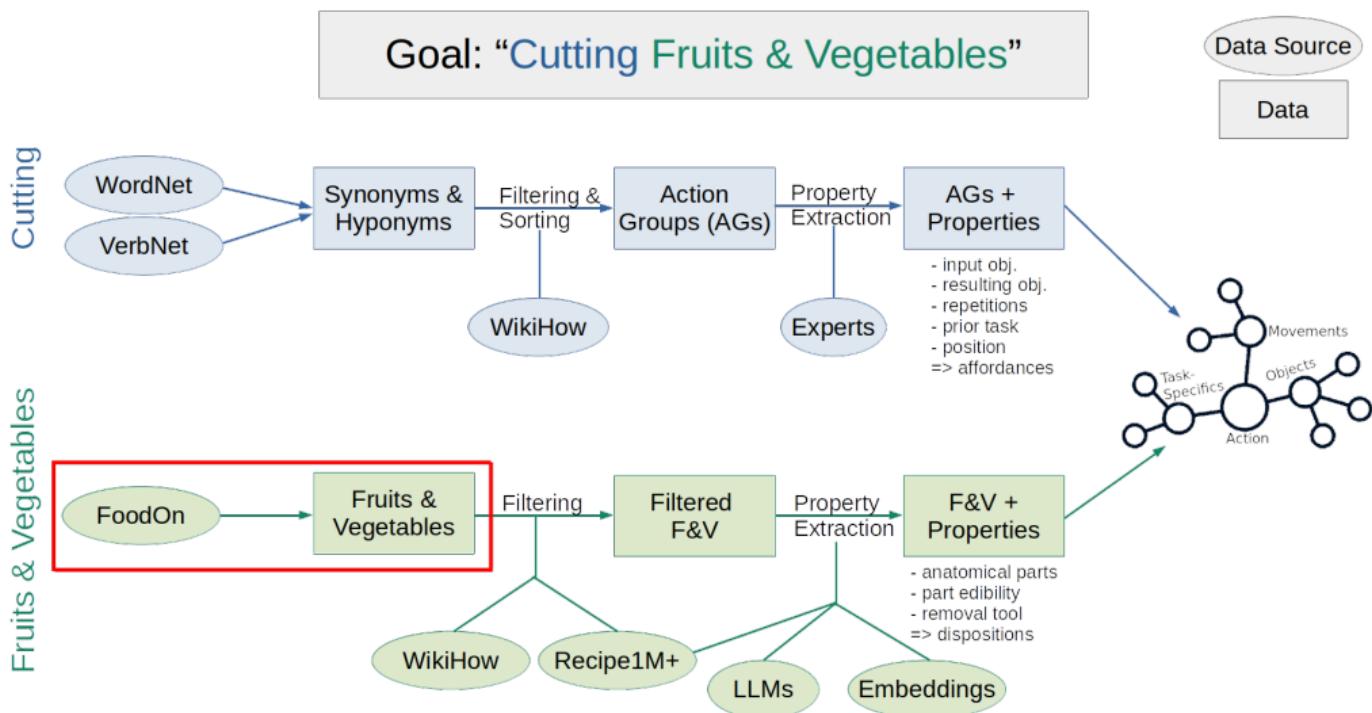
Extraction of Object Knowledge

Demonstration 03

Jupyter Notebook on Object Knowledge Extraction:

Tut-ExeWebKnow/Knowledge Extraction/object_extract.ipynb

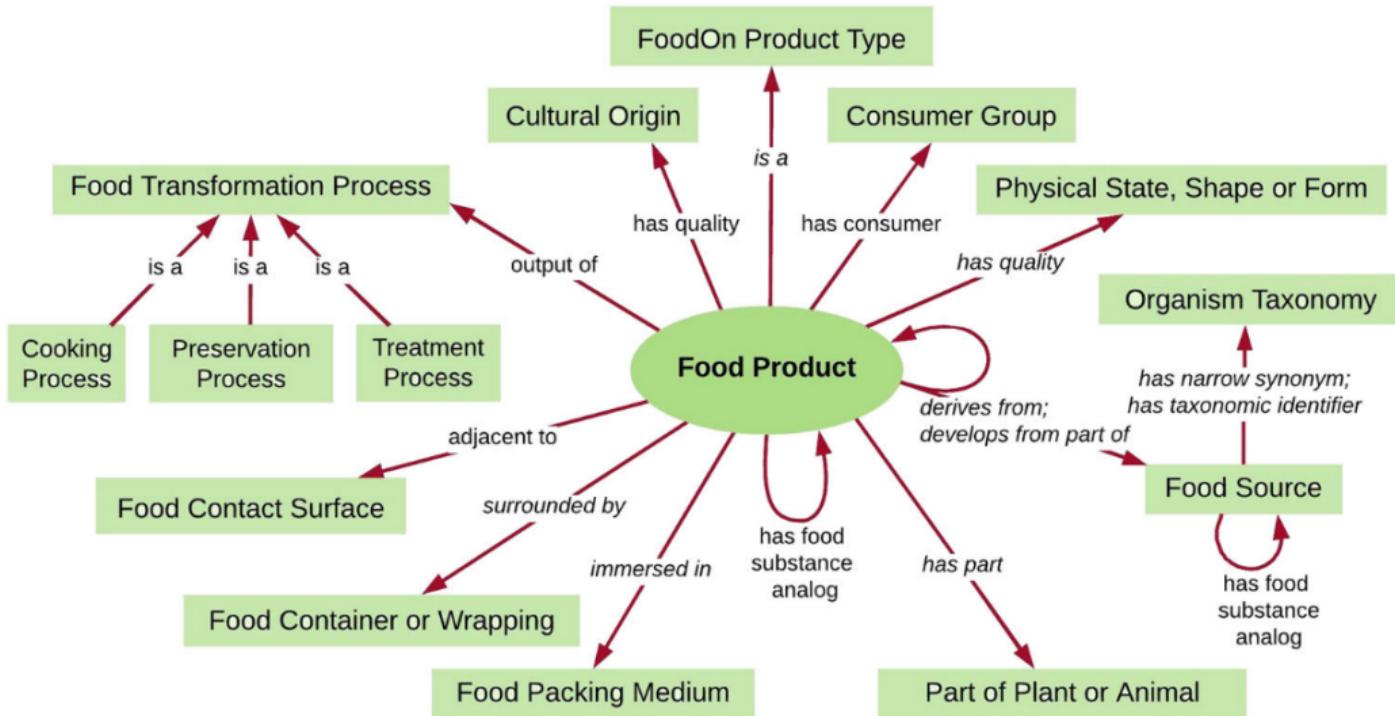
Extraction of Object Knowledge



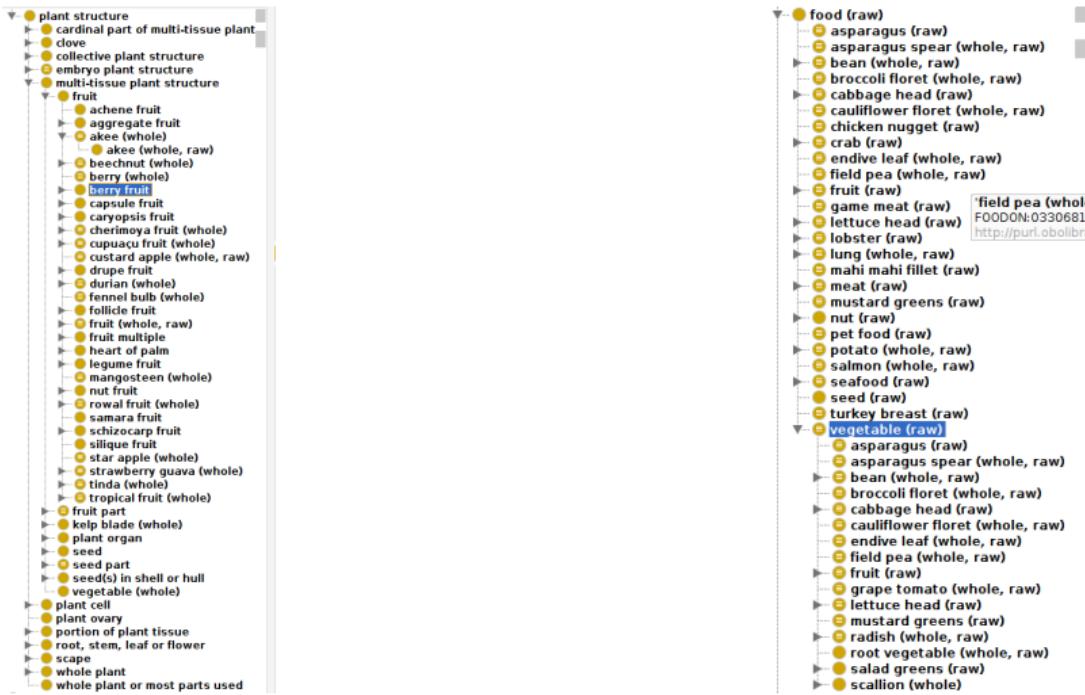
FoodOn [Doo+18] |

- Global Farm-to-Fork Ontology describing foods commonly known in cultures from around the world
- Vocabulary is based on a food indexing thesaurus (*LanguaL*)
- Covers different food-related aspects like:
 - food product terminology gaps
 - **animal and plant food sources**
 - food categories and products
 - preservation processes
 - food traceability
 - contact surfaces & packaging
 - regional variants of food products & terms (e.g. *biscuit*)

FoodOn [Doo+18] II



Fruits & Vegetables in the FoodOn



Querying FoodOn for Fruits

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX foodon: <http://purl.obolibrary.org/obo/>
SELECT ?fruit_label (SAMPLE(?fruit_id) AS ?rndm_fruit_id) (SAMPLE(?def) AS ?rndm_def)
WHERE {
    ?fruit_id rdfs:label ?label.
    ?fruit_id rdfs:subClassOf+ food:PO_0009001.
    OPTIONAL { ?fruit_id food:IAO_0000115 ?def. }
    fruit
    textual definition
    BIND (LCASE(STR(?label)) AS ?str_label).
    BIND (STRBEFORE(?str_label, "(") AS ?fruit_label).
    FILTER CONTAINS(?str_label, "whole").
    caryopsis fruit
    FILTER NOT EXISTS { ?fruit_id rdfs:subClassOf* food:PO_0030104. }
    FILTER (?fruit_id != food:FOODON_03304644).
    fruit (whole, raw)
}
GROUP BY ?fruit_label
ORDER BY ?fruit_label
```

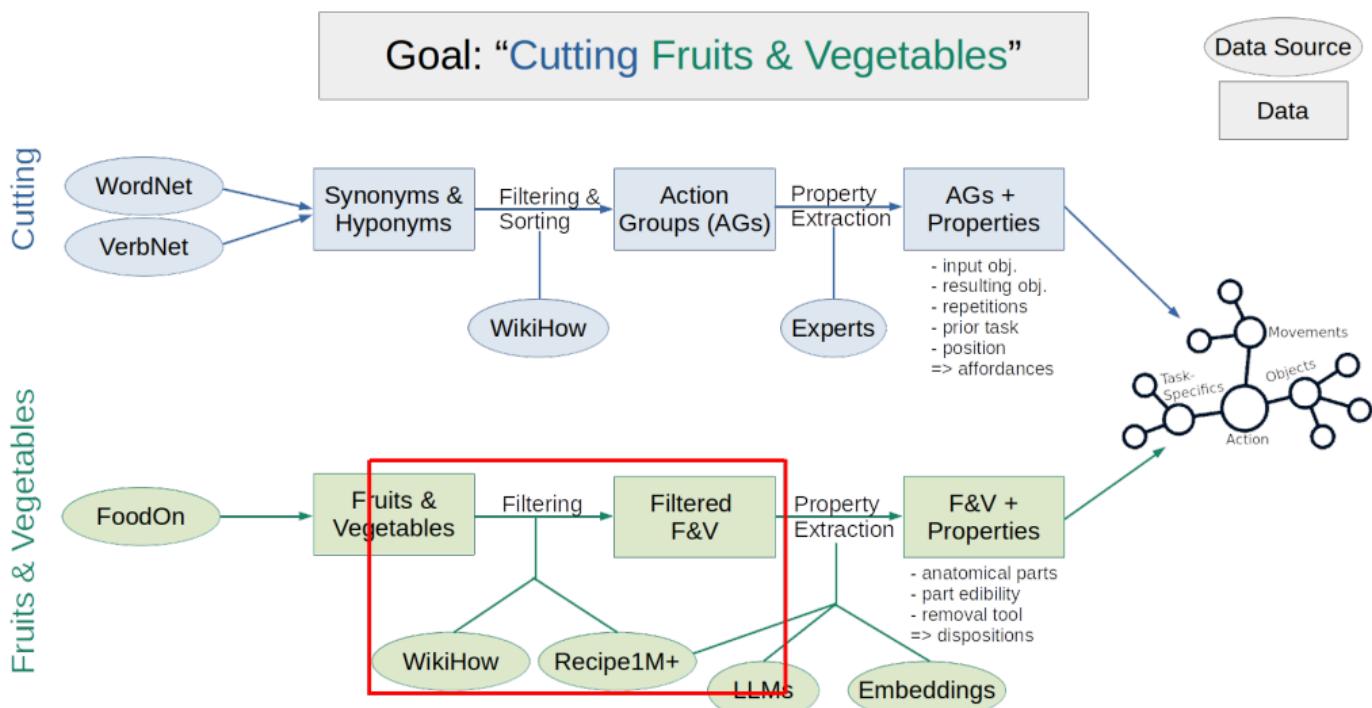
Result: 257 different fruits

Querying FoodOn for Vegetables

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX foodon: <http://purl.obolibrary.org/obo/>
SELECT ?veg_label (SAMPLE(?veg_id) AS ?rndm_veg_id) (SAMPLE(?def) AS ?rndm_def)
WHERE {
    ?veg_id rdfs:label ?label.
    ?veg_id rdfs:subClassOf+ food:FOODON_03302008, vegetable (raw)
    OPTIONAL { ?veg_id food:IAO_0000115 ?def. } textual definition
    BIND (LCASE(STR(?label)) AS ?str_label).
    BIND (STRBEFORE(?str_label, "(") AS ?veg_label).
    FILTER NOT EXISTS { ?veg_id rdfs:subClassOf* food:FOODON_03302007. }
}
GROUP BY ?veg_label
ORDER BY ?veg_label fruit (raw)
```

Result: 31 different vegetables

Extraction of Object Knowledge



Filtering Fruits and Vegetables - Motivation

- Not all 288 fruits & vegetables are equally important for Everyday cutting tasks
- Is there enough knowledge available to reason about a fruits property?
- Example: *Mangosteen* occurs in 6 out of $\geq 1.000.000$ recipes - relevant?



Filtering Fruits and Vegetables - Method

- Search for fruits & vegetables using the WikiHow Analysis Tool [Töb22] and the a tool for analyzing the Recipe1M+ corpus [Mar+21; Töb23]
- Search in all parts of the recipe & all parts of a WikiHow article
- Only include fruits/vegetables occurring in $\geq 1\%$ of any part

Filtering Fruits and Vegetables - Results

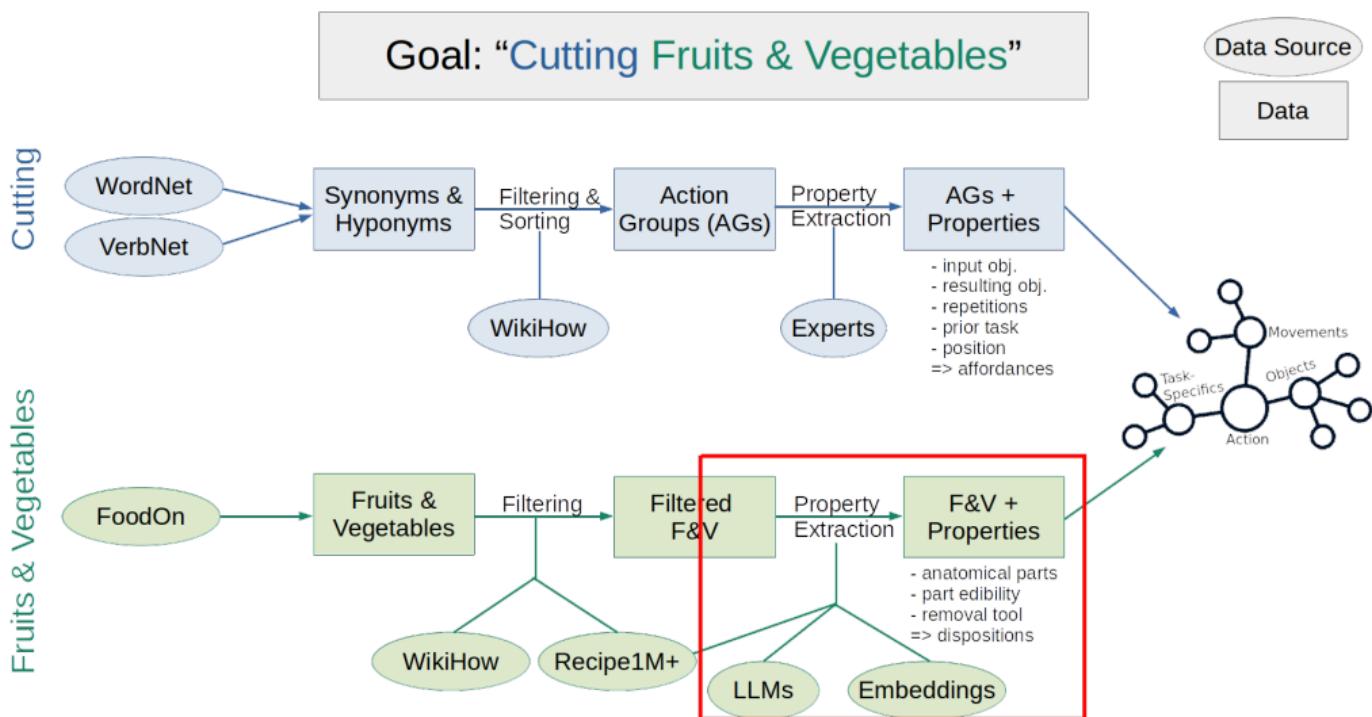
Fruits: 257 → 15

almond, apple, banana, cherry, coconut, lemon, lime, olive, orange, pepper, pineapple,
pumpkin, squash, strawberry, tomato

Vegetables: 31 → 1

bean

Extraction of Object Knowledge



Extraction of Object Properties

Relevant Properties for *Cutting*

- **Anatomical parts**

What parts that could potentially be removed exists? (e.g. peel, core, stem, shell)

- **Part edibility**

Can an anatomical part be eaten, is it unpalatable or even poisonous?

- **Removal tool**

What tool should be used to remove an unpalatable / poisonous anatomical part?
(e.g. knife, peeler)

Word Embeddings: GloVe, NASARI & ConceptNet Numberbatch

GloVe [PSM14]

- Unsupervised learning algorithm
- Trained on aggregated global word-word co-occurrence statistics
- Corpus: Wikipedia (2014) + Archive of newswire data

Word Embeddings: GloVe, NASARI & ConceptNet Numberbatch

GloVe [PSM14]

- Unsupervised learning algorithm
- Trained on aggregated global word-word co-occurrence statistics
- Corpus: Wikipedia (2014) + Archive of newswire data

NASARI [CPN15]

- Vector representation technique
- Trained on lexicographic (e.g. WordNet) as well as encyclopedic resources (e.g. Wikipedia) \Rightarrow 2 vectors per word

Word Embeddings: GloVe, NASARI & ConceptNet Numberbatch

GloVe [PSM14]

- Unsupervised learning algorithm
- Trained on aggregated global word-word co-occurrence statistics
- Corpus: Wikipedia (2014) + Archive of newswire data

NASARI [CPN15]

- Vector representation technique
- Trained on lexicographic (e.g. WordNet) as well as encyclopedic resources (e.g. Wikipedia) \Rightarrow 2 vectors per word

Numberbatch [SCH17]

- Pre-computed, multilingual embeddings
- Based on ConceptNet, Word2Vec, GloVe, a movie subtitles corpus
 \Rightarrow Focus on Commonsense Knowledge

Using the Word Embeddings for Extracting Anatomical Parts

- 1 Take one of the three word embeddings (GloVe, NASARI or ConceptNet Numberbatch)
- 2 Iterate over all fruits and anatomical parts & calculate their embedding
- 3 Calculate the cosine similarity between the vectors
- 4 Compare the cosine similarity to a threshold

Word Embeddings for Extracting Anatomical Parts - Results

Method	Acc.	Prec.	Rec.	Spec.	F1	Threshold
CN Numberbatch	.788	.609	.636	.845	.622	Cossim \geq 0.20
CN Numberbatch	.825	1.00	.364	1.00	.533	Cossim \geq 0.30
GloVe	.550	.348	.727	.483	.471	Cossim \geq 0.25
GloVe	.688	.435	.455	.776	.444	Cossim \geq 0.40
NASARI	.750	.571	.364	.897	.444	Cossim \geq 0.75
GloVe	.738	.533	.364	.879	.432	Cossim \geq 0.50
NASARI	.500	.295	.591	.466	.394	Cossim \geq 0.50

Using the Recipe Corpus for Extracting Anatomical Parts

Method 1 - Bigrams:

- 1 Provide a list of possible anatomical parts with their synonyms
- 2 Look for the co-occurrence of these terms with the fruit / vegetable by searching for their bigrams (*[fruit] [part]*)
Example: Use a melon baller to dig out the **apple core**.
- 3 Compare the amount of co-occurrences to a threshold

Using the Recipe Corpus for Extracting Anatomical Parts

Method 1 - Bigrams:

- 1 Provide a list of possible anatomical parts with their synonyms
- 2 Look for the co-occurrence of these terms with the fruit / vegetable by searching for their bigrams (*[fruit] [part]*)
Example: Use a melon baller to dig out the **apple core**.
- 3 Compare the amount of co-occurrences to a threshold

Method 2 - 2-Step Filter:

- 1 Provide a list of possible anatomical parts with their synonyms
- 2 Look for the co-occurrence of these terms with the fruit / vegetable by first searching for the fruit and then searching for the part in the results
Example: Remove a 1 inch strip of **peel** around the middle of each **apple**.
- 3 Compare the amount of co-occurrences to a threshold

Recipe Corpus for Extracting Anatomical Parts - Results

Method	Acc.	Prec.	Rec.	Spec.	F1	Threshold
2-Step Filter	.863	.824	.636	.948	.718	Occ. in $\geq 1\%$ of steps
Bigrams	.688	.463	.864	.621	.603	Occ. in any step
2-Step Filter	.738	.517	.682	.759	.588	Occ. in $\geq 0.5\%$ of steps
Bigrams	.788	.667	.455	.914	.541	Occ. in $\geq 0.1\%$ of steps

Prompting ChatGPT & GPT-4 for Anatomical Parts

System Message: Can you please answer the following question only with the existing parts and without any additional text.

User Message: Which of the following four food parts do you think are part of a [fruit]:
A core, a shell, a peel and a stem.

Comparing Different Methods for Extracting Anatomical Food Parts

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

Food Part Edibility Using LLMs

System Message: Can you please answer the following question only with the chosen edibility and without any additional text.

User Message: What is the edibility for the [part] of a/an [fruit]?

You can choose between three possibilities: Edible, Must Be Avoided or Should Be Avoided

Food Part Edibility Using LLMs

System Message: Can you please answer the following question only with the chosen edibility and without any additional text.

User Message: What is the edibility for the [part] of a/an [fruit]?

You can choose between three possibilities: Edible, Must Be Avoided or Should Be Avoided

Method	Prec.	Rec.	F1
GPT-4	.613	.542	.410
ChatGPT	.195	.444	.253

Removal Tool Extraction Using LLMs

System Message: Can you please answer the following question only with the 1 tool you think works best and without any additional text.

User Message: What tool would you use to remove the [part] of a/an [fruit]?

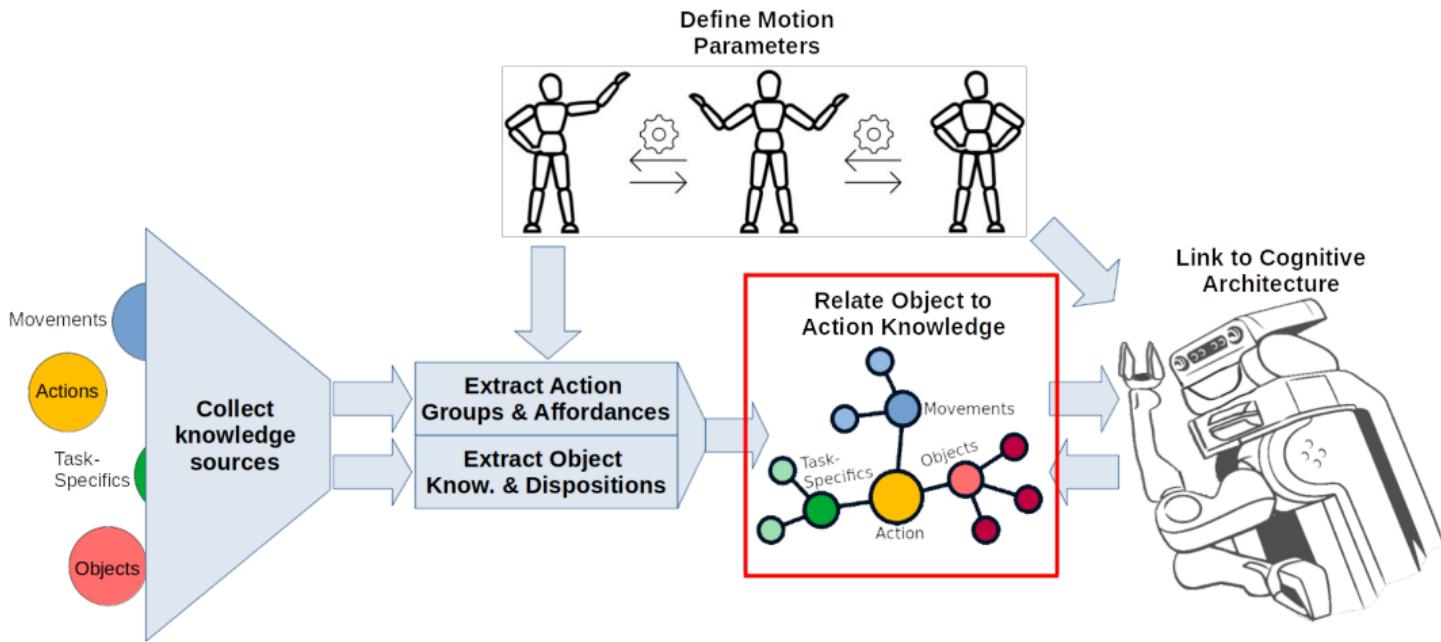
System Message: Can you please answer the following question only with the 1 tool you think works best and without any additional text.

User Message: What tool would you use to remove the [part] of a/an [fruit] if you would have to choose between a knife, a nutcracker, a hand, a spoon or a peeler?

Comparing LLMs Regarding Removal Tools

Method	Prec.	Rec.	F1
GPT-4 (Closed)	.798	.621	.577
ChatGPT (Closed)	.583	.505	.434
GPT-4 (Open)	.790	.446	.428
ChatGPT (Open)	.374	.408	.341
Recipe1M+ 2-Step (Complete)	.400	.383	.318
Recipe1M+ Bigrams (Fruit Only)	.269	.267	.154
Recipe1M+ 2-Step (Fruit Only)	.225	.233	.102

Knowledge Engineering Methodology: Step 4



Knowledge Engineering Methodology: Step 4

Connect affordances to dispositions in the TBox using `affordsTask`,
`affordsTrigger` and `hasDisposition` relations introduced in the SOMA ontology:
Disposition: Cuttability of an apple

```
hasDisposition some
(Cuttability
and (affordsTask some CuttingAction)
and (affordsTrigger only (classifies only Knife)))
```

Knowledge Engineering Methodology: Step 4

Connect affordances to dispositions in the TBox using `affordsTask`,
`affordsTrigger` and `hasDisposition` relations introduced in the SOMA ontology:
Disposition: Peelability of a banana

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

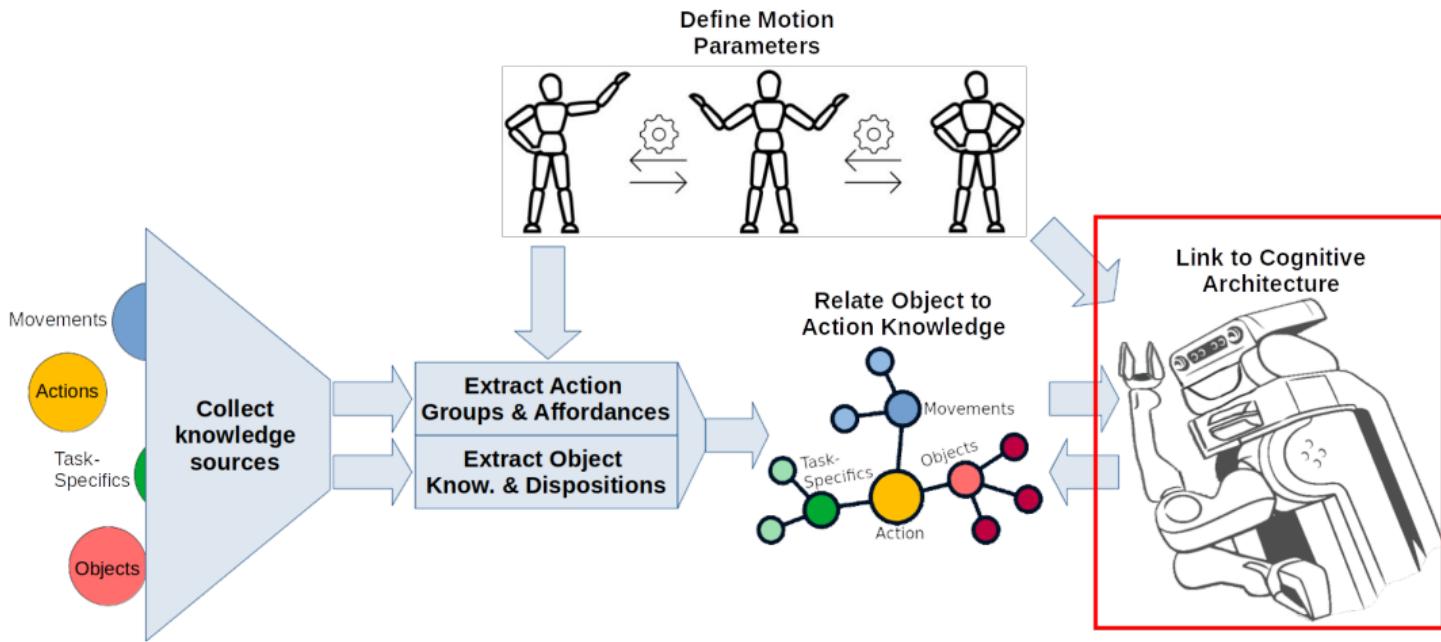
Knowledge Engineering Methodology: Step 4

Affordance and Disposition

- Link a food object to a disposition
- Dispositions can afford different tasks and tools

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

Knowledge Engineering Methodology: Step 5

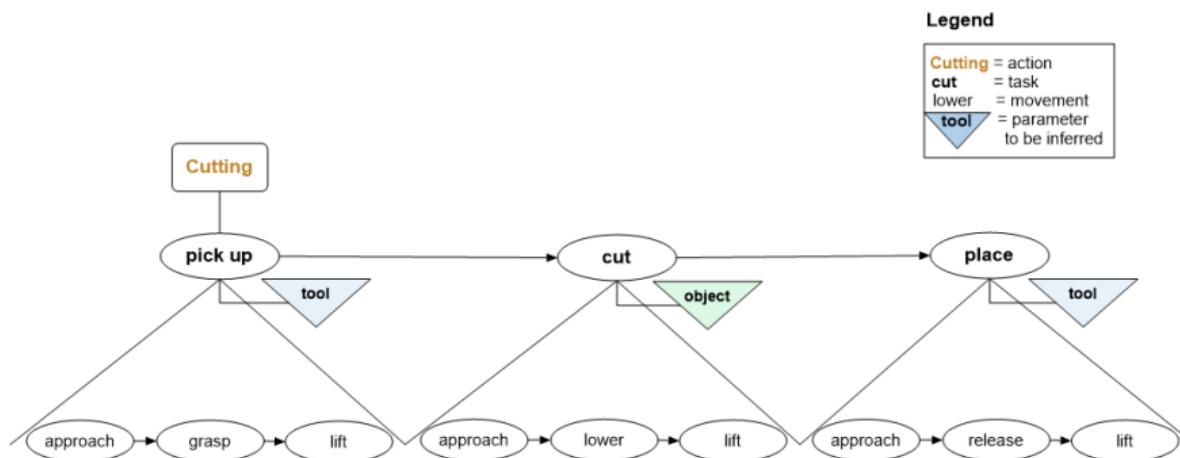


Knowledge Engineering Methodology: Step 5

Inspect Action Plan

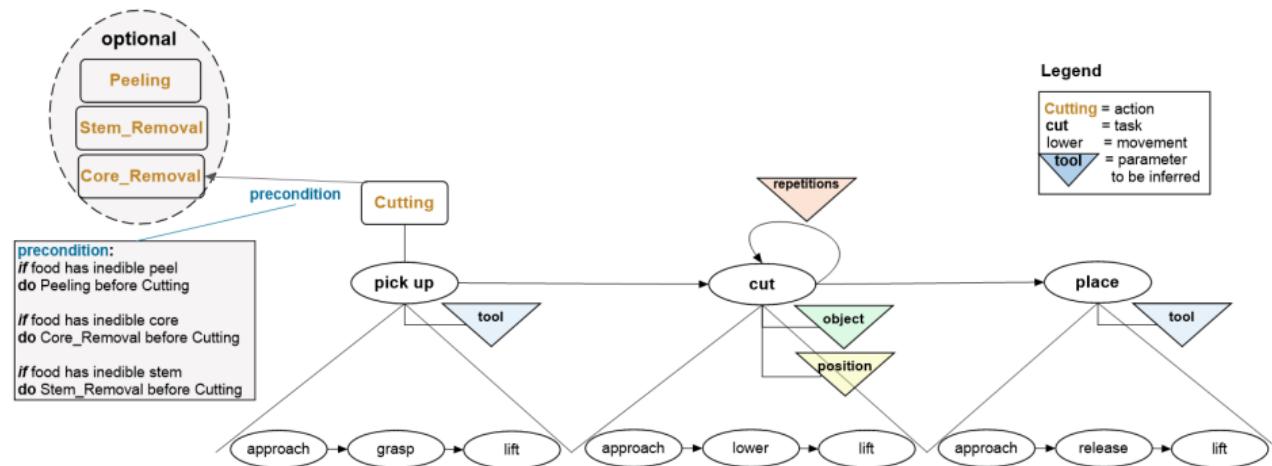
With the given plan

- a robot can cut a given object once at a set position
- Use action groups and analyzed action parameters to modify action plan



Knowledge Engineering Methodology: Step 5

Adapted action plan [Küm+24a]:



Knowledge Engineering Methodology: Step 5

- Map concepts in the generalised plan to their representation in the ontology
- Ground objects & their properties in the perception system
- Cognitive architecture needs to be able to query the ontology to gather specific parameterisations at runtime

Knowledge Engineering Methodology: Step 5

A look at action designator refinement

Refining an object type through known parameters/variables (here :knife)

(an object
(type :knife))

Knowledge Engineering Methodology: Step 5

A look at action designator refinement

Refining an action type through known parameters/ variables (here :cutting)

(an action
(type :cutting))

Knowledge Engineering Methodology: Step 5

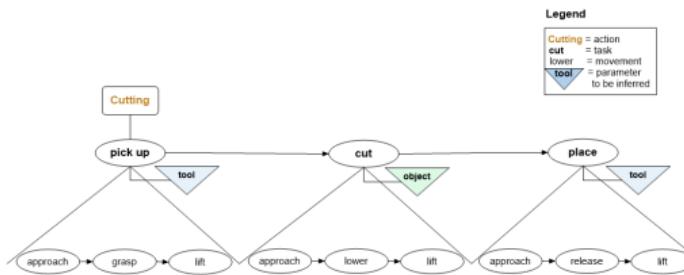
A look at action designator refinement

Refining an action type through queryable parameters/ variables (here
?navigation-goal, set in code)

```
(let ((?navigation-goal (make-pose "map" '(0.0 1.0 0.0) '(0.0 0.0 (* pi 0.0)))))  
  (perform (an action  
            (type :going)  
            (target (a location  
                     (pose ?navigation-goal)))))))
```

Knowledge Engineering Methodology: Step 5

Initial cutting action designator



```
(perform
  (an action
    (type :cutting)
    (?object_acted_on (an object (type :food) (pose ...) ...))
    (?object (an object (type :tool) (pose ...) ...))
    (?arm (a body_part (type :arm) (pose ...) ...))
    (?arm (a body_part (type :arm) (pose ...) ...)))
    (?goal (:success (:cutting (on :food))))))
```

Knowledge Engineering Methodology: Step 5

```
(perform
  (an action
    (type :cutting)
    (?object_acted_on (an object (type :food) (pose ...) ...))
    (?object (an object (type :tool) (pose ...) ...))
    (?arm (a body_part (type :arm) (pose ...) ...))
    (?arm (a body_part (type :arm) (pose ...) ...))
    (?goal (:success (:cutting (on :food))))))
```

- for all queryable variables (e.g. **?object_acted_on**)
 - first try to resolve internally (if used before)
 - then query knowledge base for info
 - try to resolve through sensors
 - assign

Knowledge Engineering Methodology: Step 5

Adapted cutting action designator

```
(perform
  (an action
    (type :cutting (?repetitions (:boolean)))
    (?prior_action (an action (type :preparing)))
    (?depends_on_task (an action (type :cutting)))
    (?object_acted_on (an object (type :food) (pose ...) ...))
    (?object (an object (type :tool) (pose ...) ...))
    (?arm (a body_part (type :arm) (pose ...) ...))
    (?arm (a body_part (type :arm) (?position (on :food)) ...))
    (?goal (:success (:cutting (on :food))))))
```

Knowledge Engineering Methodology: Step 5

```
(perform
  (an action
    (type :cutting (?repetitions (:boolean)))
    (?prior_action (an action (type :preparing)))
    (?depends_on_task (an action (type :cutting)))
    (?object_acted_on (an object (type :food) (pose ...) ...))
    (?object (an object (type :tool) (pose ...) ...))
    (?arm (a body_part (type :arm) (pose ...) ...))
    (?arm (a body_part (type :arm) (?position (on :food)) ...))
    (?goal (:success (:cutting (on :food))))))
```

Things to query from the knowledge base

- **?repetitions**
- **?prior_action**
- **?depends_on_task**
- **?position**, i.e. cutting position

Knowledge Engineering Methodology: Step 5

Prior to action execution, the robot will pose these queries:

- What tool shall be used? (unless already known)
- **?prior_action:** Are prior actions (e.g. peeling) required?
- **?depends_on_task:** Does the task depend on tasks that need to be executed (e.g. halving before quartering)?
- **?repetitions:** How many repetitions of this task need to be performed?

Knowledge Engineering Methodology: Step 5

Before cutting motion execution, the robot will pose these queries:

- **?position:** What is the needed cutting position (center or end)?
- **?object:** What is the input object to cut (whole food or food part)?
- What is the result object? This is needed to determine successful action execution

Knowledge Engineering Methodology: Step 5

Let us now look at the queries in the last jupyter notebook

Thank you for your attention!

Questions?

References I

- [BG80] Marc H. Bornstein and James J. Gibson. “The Ecological Approach to Visual Perception”. In: *The Journal of Aesthetics and Art Criticism* 39.2 (1980), p. 203. ISSN: 00218529. DOI: 10.2307/429816. JSTOR: 10.2307/429816. (Visited on 12/14/2022).
- [Tur92] Michael T. Turvey. “Ecological Foundations of Cognition: Invariants of Perception and Action.”. In: *Cognition: Conceptual and Methodological Issues*. Ed. by Herbert L. Pick, Paulus Willem van den Broek, and David C. Knill. Washington: American Psychological Association, 1992, pp. 85–117. ISBN: 978-1-55798-165-3. DOI: 10.1037/10564-004. (Visited on 12/14/2022).
- [Mil95] George A Miller. “WordNet: A Lexical Database for English”. In: *Communications of the ACM* 38.11 (1995), pp. 39–41. DOI: 10.1145/219717.219748.

References II

- [BFL98] Collin F. Baker, Charles J. Fillmore, and John B. Lowe. “The Berkeley FrameNet Project”. In: *Proceedings of the 36th Annual Meeting on Association for Computational Linguistics* -. Vol. 1. Montreal, Quebec, Canada: Association for Computational Linguistics, 1998, p. 86. DOI: [10.3115/980845.980860](https://doi.org/10.3115/980845.980860). (Visited on 09/25/2023).
- [Jai+05] Pankaj Jaiswal et al. “Plant Ontology (PO): A Controlled Vocabulary of Plant Structures and Growth Stages”. In: *Comparative and Functional Genomics* 6.7-8 (2005), pp. 388–397. ISSN: 1531-6912, 1532-6268. DOI: [10.1002/cfg.496](https://doi.org/10.1002/cfg.496). (Visited on 09/25/2023).
- [Sch05] Karin Kipper Schuler. “VerbNet: A Broad-Coverage, Comprehensive Verb Lexicon”. PhD thesis. University of Pennsylvania, 2005.

References III

- [PSM14] Jeffrey Pennington, Richard Socher, and Christopher Manning. “Glove: Global Vectors for Word Representation”. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, 2014, pp. 1532–1543. DOI: 10.3115/v1/D14-1162. (Visited on 06/01/2023).
- [ASS15] Robert Arp, Barry Smith, and Andrew D. Spear. *Building Ontologies with Basic Formal Ontology*. Cambridge, Massachusetts London, England: The MIT Press, 2015. ISBN: 978-0-262-52781-1.

References IV

- [CPN15] José Camacho-Collados, Mohammad Taher Pilehvar, and Roberto 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>.
- [PG16] Valentina Presutti and Aldo Gangemi. "Dolce+ D&S Ultralite and Its Main Ontology Design Patterns". In: *Ontology Engineering with Ontology Design Patterns: Foundations and Applications*. Ed. by Pascal Hitzler et al. Studies on the Semantic Web 25. Berlin, Germany: AKA GmbH Berlin, 2016, pp. 81–103. ISBN: 978-3-89939-715-6.

References V

- [SCH17] Robyn Speer, Joshua Chin, and Catherine Havasi. "ConceptNet 5.5: An Open Multilingual Graph of General Knowledge". In: *AAAI 31.1* (Feb. 2017). ISSN: 2374-3468, 2159-5399. DOI: [10.1609/aaai.v31i1.11164](https://doi.org/10.1609/aaai.v31i1.11164). (Visited on 01/16/2023).
- [CLW18] Richard Crang, Sheila Lyons-Sobaski, and Robert Wise. "Fruits, Seeds, and Seedlings". In: *Plant Anatomy : A Concept-Based Approach to the Structure of Seed Plants*. Cham: Springer International Publishing Imprint: Springer, 2018, pp. 649–678.
- [Doo+18] Damion M. Dooley et al. "FoodOn: A Harmonized Food Ontology to Increase Global Food Traceability, Quality Control and Data Integration". In: *npj Sci Food* 2.1 (Dec. 2018), p. 23. ISSN: 2396-8370. DOI: [10.1038/s41538-018-0032-6](https://doi.org/10.1038/s41538-018-0032-6). (Visited on 08/10/2022).

References VI

- [Mar+21] Javier Marín et al. “Recipe1M+: A Dataset for Learning Cross-Modal Embeddings for Cooking Recipes and Food Images”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43.1 (Jan. 2021), pp. 187–203. ISSN: 1939-3539. DOI: [10.1109/TPAMI.2019.2927476](https://doi.org/10.1109/TPAMI.2019.2927476).
- [Ahn+22] Michael Ahn et al. *Do As I Can, Not As I Say: Grounding Language in Robotic Affordances*. Aug. 2022. DOI: [10.48550/arXiv.2204.01691](https://doi.org/10.48550/arXiv.2204.01691). arXiv: 2204.01691 [cs]. (Visited on 12/16/2022).

References VII

- [Beß+22] Daniel Beßler et al. “Foundations of the Socio-physical Model of Activities (SOMA) for Autonomous Robotic Agents”. In: *Formal Ontology in Information Systems*. Vol. 344. Frontiers in Artificial Intelligence and Applications. Amsterdam: IOS Press, 2022, pp. 159–174. arXiv: 2011.11972 [cs]. URL: <https://ebooks-iospress.nl/doi/10.3233/FAIA210379> (visited on 07/25/2022).
- [Töb22] Jan-Philipp Töberg. *WikiHow Instruction Analysis for Robot Manipulation*. GitHub, 2022. URL: <https://github.com/Food-Ninja/WikiHow-Instruction-Extraction>.

References VIII

- [Lia+23] Jacky Liang et al. “Code as Policies: Language Model Programs for Embodied Control”. In: *40th IEEE International Conference on Robotics and Automation (ICRA)*. London, UK: IEEE, 2023, pp. 9493–9500. DOI: [10.1109/ICRA48891.2023.10160591](https://doi.org/10.1109/ICRA48891.2023.10160591). (Visited on 02/22/2023).
- [Töb23] Jan-Philipp Töberg. *Recipe Analysis for Robot Manipulation*. GitHub, 2023. URL: <https://github.com/Food-Ninja/RecipeAnalysis>.
- [Küm+24a] Michaela Kümpel et al. “[UNDER REVIEW] Cut, Chop, Slice or Dice: Parameterising General Action Plans Using Knowledge Graphs”. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2024)*. Abu Dhabi, UAE, 2024.

References IX

- [Küm+24b] Michaela Kümpel et al. “Towards a Knowledge Engineering Methodology for Flexible Robot Manipulation in Everyday Tasks”. In: *International Workshop on Actionable Knowledge Representation and Reasoning for Robots (AKR³)*. Heraklion, Crete, Greece, 2024.