Technische Universitaet Muenchen - M.Sc. Informatics

InterDisziplinaeres Project(IDP):

Probabilistic Robot Action Cores - uncertainty verbalization for semantic disambiguation

Institute of Automatic Control Engineering
Department of Electrical Engineering and Information Technology
Examiner: Prof. Joerg Conradt - Advisor: Daniel Nyga, M.Sc.
Student: Nicholas H. Kirk - mat. 03624396

keywords: cognitive architectures, computational linguistics, robotics.

Contents

1 introduction

1.1 to the project

A more detailed definition of the preexistent background is given in the scope definition (for a high level overview) and in Nyga12actioncore quote (for a technical deep dive).

1.2 to the document

Given the interdisciplinarity of the topic (i.e. cognition, natural language processing, statistical relational learning) this document is intended to describe the undergone research process as close as possible, both from a technical and ideation standpoint, by following the incremental progress of the project.
An overview of the contribution to the background will be given, to then progressively shift towards more technical and linguistical aspects of the implementation.

2 preliminary understanding

abstraction level Probabilistic Robot Action Cores are an everyday-task oriented formalization that abstracts from the Markov Logic Network formulas at a lower level. We therefore focus on the verbalization of uncertain elements in the PRAC serialization. The expanded MLN statements are not of direct interest for the scope, since their verbalization would not be of interest and scarcely intelligible.

2.1 interaction overview

Let us lay the ground for our interaction analysis:

$$\boxed{\begin{array}{lll} & \text{Human} \rightarrow \text{Robot} & \textit{instructing} \\ \textit{0} \ldots n & \begin{cases} \text{Robot} \rightarrow \text{Human} & \textit{querying} \\ \text{Human} \rightarrow \text{Robot} & \textit{clarifying} \end{cases} & \end{array}}$$

Abstract interaction overview

More specifically:

1. Human →Robot: NaturalLanguageInstruction (English language)
2. Robot→Human: AmbiguityResolutionRequest (defined subset of the English language ⇔ with PRAC MLN)
3. Human→Robot: NaturalLanguageDisambiguation (English Language)

2.2 linguistic concerns

We now analyse the previously stated points, highlighting possible concerns strictly related to the interaction:

1. Normally in imperative form. In English we have more ambiguity given that imperative form can lead to loss of correct use of grammar (in particular when the NLI is long), but we have lower parsing complexity considering there is no verb conjugation (the infinite form is used).
2. Our cognitive system will verbalize a sentence in which asks for clarifications regarding a set of aspects related to the task, using the lexicon learnt in (1), with which the relationships of the PRAC has been built.
3. The human will provide an explanation, possibly providing solutions to all ambiguity concerns of the robot, in a non-imperative and maybe hypotactical manner (with subordinate sentences).

vocabulary variation Mapping between lexicon of (1), (2) and (3) can be complex (e.g. First it is instructed as “X needs Y“, then “X requires Y and Z“). Between (1) and (2) is trivial, but the correlation with (3) is not. Further questioning could hypothetically lead to loops of lexicon clarification. The system should foresee these possible changes and verify synonyms when importing.

anaphora resolution Correct anaphora resolution of informal sentences (e.g.”The pancake needs flower, it can be found on the shelf“, it refers to flower).

2.3 possible disambiguation classes

In this section we try to identify what kind of clarification could be necessary are possible or likely, formalizing a list of possible clarification classes that we might need to verbalize.

Taxonomy clarification Classification of an object that is useful in the scope, clarifying the higher class.
(1) Querying: “What is X¿‘
(2) Clarifying: “X is a type of flower.“

Temporal clarification Most frequently a time instruction is ambiguous. A post processing phase will be necessary to classify the residual ambiguity (i.e. ”for dinner“).
(1) Instructing: “Flip the pancake in a while.“
(2) Querying: “When should the pancake be flipped¿‘

Anaphora resolution References to previous sentences could be present in an anaphoric form.

(1) Clarifying: "Cook it now."
(2) Querying: "What is 'it'¿'

Context adaptation The NLI could ask for something that is not feasible in that context (e.g. the object that should be used is either not present or not useful for the task).

(1) Instructing: "Flip the pancake."
(context: no spatulas or similars around)
(2) Querying: "What can flip the pancake¿'

(1) Instructing: "Flip the pancake with a screwdriver."
(context: spatula around, presence of screwdriver irrelevant)
(2) Querying: "The spatula can be used¿'

Identity clarification Another when multiple instances of artificial assistants are present - it could then be useful to disambiguate who will accomplish a specific task. "Who is the robot that should cook the pancake¿'
(1) Instructing: "Flip the pancake."
(context: multiple assistants)
(2) Querying: "Who should flip the pancake¿'

3 reuse of state-of-the-art technologies

3.1 Controlled Language - ACE

Attempto Controlled English (ACE) is a controlled natural language, i.e. a subset of standard English with a restricted syntax and a restricted semantics described by a small set of construction and interpretation rules.[1]

ACE can serve as knowledge representation, specification, and query language, and in our context it is used as a linguistic notation to interface humans and peer robotic assistants, without ambiguity - Though ACE appears perfectly natural and indistinguishable from English, it is in fact a formal language.

ACE limitations

workaround to personal pronouns since these in first person singular are purpusely not implemented in ACE and the other forms can incur in ambiguous anaphora refences, we will use the passive form, even if this brings a partial loss of semantics (e.g.*"I will boil the water" →"The water will be boiled."*).

3.2 Metalanguage: Discourse Representation Structures (DRS)

Discourse representation structures (DRS) represent a hearer's mental representation of a discourse as it unfolds over time. There are two critical components to a DRS:

- A set of DRS referents representing entities which are under discussion.
- A set of DRS conditions representing information that has been given about discourse referents.

Example:
(1) A robot flips a pancake.
The DRS of **(1)** can be notated as **(2)** below:

(2) [x,y: robot(x), pancake(y), flips(x,y)]

In which x,y are discourse referents, and the remaining statements are discourse conditions. To show sequentiality, we add the following sentence to the discourse.
(3) A robot flips a pancake. He burns it.
(4) [x, y: robot(x), pancake(y), flips(x,y), burns(x,y)]

We shall make use of DRS as the verbalizer's intermediate representation, specifically the DRS implementation of the Attempto Project.

Why not OWL ontologies? OWL ontologies are a defacto standard and currently standard drafts of many versions are under examination, however we observe the following theoretical limitations:

- OWL is not easily human-readable, and a verbalization would require postprocessing.
- OWL cannot express modal formulations (these being likelihood, ability, permission, and obligation), since these imply possibility and not certainty, and this conflicts directly with the concept of a non-probabilistic ontology.
- OWL cannot express questions. This can be overid by mild postprocessing on a question that treats a query word (e.g. when, what, who) as a noun.

3.3 reasons behind the technology adoptions

A preexistent language framework able to verbalize relationships between entities is necessary given its ability to conjugate verbs, resolve references, set appropriate gender and plurality declinations. The use of a specific implementation of Controlled Language is desirable for compatibility between future ontologies.

4 scenario development (technical)

The process loops n times to clarify ambiguity, until this is reduced to a minimal preestablished level (n can be bounded for hindrance reasons i.e. excessive questioning).

$$\begin{array}{rl} \text{NLI} \rightarrow \text{PRAC} & \textit{instructing} \\ 0\ldots n \begin{cases} \text{PRAC} \rightarrow \text{DRS} \rightarrow \text{ACE} \\ \text{NLD} \rightarrow \text{PRAC inf.merge} \end{cases} & \begin{array}{l}\textit{querying}\\ \textit{clarifying}\end{array} \end{array}$$

High-level technical overview

Technical modules required:

1. Prioritization module and assessment of probabilities [RNF.tech.1]
2. DRS constructor module [RNF.tech.2]
3. DRS to ACE module [RNF.tech3]
4. Core to call the prioritization module, consult config files (global and local), verbalize (i.e. call the DRS constructor, call the DRS>ACE module).

importance of syntactic relationships For each specific instance there is a set of syntactic relationships for each action role, information that is necessary for the language generation of the questions.

Upon construction of a specific PRAC definition, a Controlled Language sentence will be defined as one of their possible syntactic mappings. This is possible This structure mapping is partially obtainable from the parsing phase done via the 'stanford parser' 3rd party software module. The rest of the structure is related to the syntactic roles of the understated PRAC part.

Why not formalize the NLI in CL? The act of mapping the syntactic analysis of the hole NLI and the understated part to DRS atoms is equivalent to a full logic formalization of such natural language statements (bringing the NLI from natural to deterministic), work that is to-date considered complex and postponed to future research.

4.1 action role querying

Once these features are indentified, preestablished DRS templates can be used as metalevel for the generation.

We now examine the specific case in which one or more of the action roles have not been assigned.

To do so, we have to retrieve the syntactic role(s) of the badly parsed action role(s) together with other lemmas that share 'close' syntactic relationships. From that we proceed in generating the question querying the concept and defining the 'close' context.

4.2 answer integration

4.3 context constraining and identification

We require a measure of 'semantic closeness' for testing the semantic coherence:

- of elements of the NLI
- of inferred elements
- of incoming clarifications

A metric that compounds semantic measures of the lemmas of the NLI is paramount for context identification.

Traditional or novel approaches via semantic distance measures do not seem feasible, given what are considered to be to-date limitations in our setting of WordNet, our source for both lexicon and semantics.

We require a measure that comprises the notions of:

- Taxonomical information
- active or passive modality attributes (e.g. '-can-be-eaten' or '-can-eat')

modal attribute derivation We present a novel use of syntactic manipulation for meaning retrieval. Precisely, we operate syntactic tag analysis on a corpora built on previously prac-inferred data, obtaining set of modal relationships between semantically tagged nouns.

Intuitively, the given set of natural language instructions coming from human expression can be considered 'reliable', therefore it follows that a correct meaning interpretation together with the word ordering can also provide information regarding modality. (example: humans regard something as feasible if they have already hear about it or seen it).

Example:

The spatula flips the pancake. **The spatula:*can-flip*, the pancake:*can-be-flipped***

corpora analysis: S-V-O tuples In order to not suffer from misleading bad syntactic parsing, we restrict the analysis to only S-V-O tuples, that are proven to be quite reliably identified.

For this analysis, we maintain the higher-level assumptions that had already been made, for instance anaphora resolution heuristics.

Why doesn't wordnet fulfill the requirements?

- only taxonomical information is of use: no modality attributes present
- lack of unambiguous semantics (concepts are expressed in natural language and not formally)
- similar meanings among entries (eccessively fine-grained or varied duplicates)
- lack of a generalized criterion for distinguishing entries of a same word

Children's tests The system should identify correctly the context of our sentence, act that is trivial for humans, even in young age. To exemplify we refer to the following children's game.:
find the *intruder* term: (horse dog rabbit subway)

4.4 badly inferred components

We now examine the case in which some understated aspects were not inferrable with confidence.

.1 verbalization example

Example 1: serialized PRAC of the verb 'Filling'

```
action_core: Filling
inherits_from: ContainerFocusedPlacing
examples:
    - The GOAL has to be filled with the THEME that is from the SOURCE.

action_roles:
    - Goal:
        - definition: A goal is an area that has to be filled .
    - Theme:
        - definition: The Theme is the physical object
          substance which changes location.
    - Source:
        - definition:
```

We now analyse a specific instance, making use of the shown PRAC:

NaturalLanguageInstruction: "The pan has to be filled with the OliveOil that is from the Cupboard."
In the following we exercise the questioning for the GOAL, THEME and SOURCE respectively:

Example 2: ACE questions and resulting DRS

```
(1) What has to be filled with the OliveOil?
(2) The pan has to be filled with what?
(3) The OliveOil is from what?

DRS =
[]
   QUESTION
   [A]
   query(A,what)-1/1
      MUST
      [B,C]
      property(B, filled ,pos)-1/7
      predicate(C,be,A,B)-1/
      modifier_pp(C,with,named(OliveOil))-1/8
   QUESTION
   [D]
   object(D,pan,countable,na,eq,1)-2/4
      MUST
      [E,F,G]
      query(E,what)-2/12
      property(F, filled ,pos)-2/10
      predicate(G,be,D,F)-2/
      modifier_pp(G,with,E)-2/11
   QUESTION
   [H,I]
   query(H,what)-3/5
   predicate(I,be,named(OliveOil))-3/3
   modifier_pp(I,from,H)-3/4
```