Controlled Natural Languages for Language Generation in Artificial Cognition

Nicholas H. Kirk[1], Daniel Nyga[1,2], Michael Beetz[2]

[1]Intelligent Autonomous Systems, Technische Universität München, Germany

[2]Institute for Artificial Intelligence & TZI, University of Bremen, Germany

nicholas.kirk@tum.de, nyga@cs.tum.edu, beetz@cs.uni-bremen.de

Abstract—In this paper we discuss, within the context of artificial assistants performing everyday activities, a resolution method to disambiguate missing or not satisfactorily inferred action-specific information via explicit clarification. While arguing the lack of preexisting robot to human linguistic interaction methods, we introduce a novel use of Controlled Natural Languages (CNL) as means of output language and sentence construction for doubt verbalization. We additionally provide implemented working scenarios, state future possibilities and problems related to verbalization of technical cognition when making use of Controlled Natural Languages.

I. INTRODUCTION

In everyday routine activities, robotic assistants and co-workers will have to perform a variety of tasks for which they cannot be pre-programmed because they are not known at production time. Research in the field of cognitive robotics envisions service robots that autonomously acquire new skills and adapt existing ones to new tasks and environments. They must decide on *how* to perform a particular activity at runtime, which requires them to infer the *appropriate* actions to be executed on the *appropriate* objects in an *appropriate* way. They have to perform what is commonly referred to as *everyday activity*, which has been proven to be a very knowledge-intensive task [1], [2]. It requires context awareness and flexibility in action parametrization when operating in real world settings with partially available information, which is referred to as the "open world challenge".

Recent research in the field of cognitive robotics aims to make knowledge sources available for robots, which have been created by humans and are intended for human use. For some domains such as daily household tasks (e.g. cooking, cleaning up), step-by-step plans and recipes from web pages like *wikihow.com* have been successfully used for feeding such common sense knowledge about actions and objects into knowledge bases of mobile robotic platforms and for transforming such recipes into executable robot plans [3].

However, as these recipes are presented in natural language, severe ambiguity, vagueness and underspecification have been identified as major challenges in translating such specifications into plans, since many missing key pieces of information are generally considered common sense to the human. As an example, consider the natural-language instruction "Flip the pancake", taken from a recipe for making pancakes: In order to perform the action successfully, a robot needs for instance to decide which utensil to use
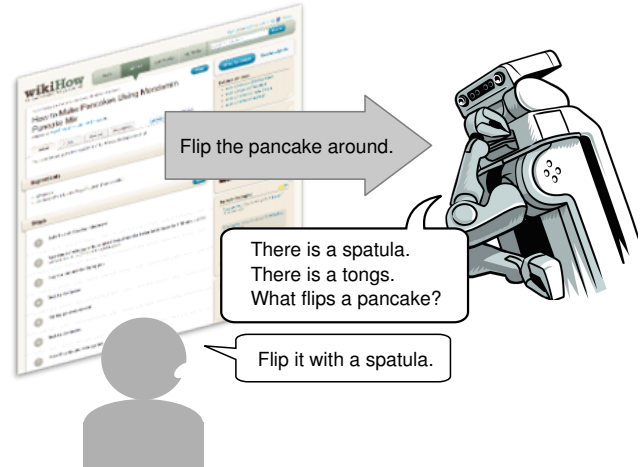

Fig. 1.   Representation of a disambiguation interaction

(e.g. a spatula), where to hold it (e.g. at its handle) and what part of it to put underneath the pancake (e.g. the blade), and where to flip it from (e.g. the burner). Current research in cognitive robotics [4] aims to build action verb-specific knowledge bases that fill these knowledge gaps and enables a robot to infer the information which is *needed* in order to perform a particular activity, based on what is *given* in a naturalistic action specification.

However, such inference might be insufficient to formulate action specification and require the robot to fall back on human assistance. As an example, consider a situation where the robot is asked to flip a pancake, but the knowledge base does not contain sufficient information about what instrument is to be used (e.g. it has no strong preference for a spatula over barbecue tongs). In such a case, the robot has to explicitly ask a human for instrument clarification. Fig. 1 illustrates such a situation. Taxonomical and compositional relationships or object role understanding are only some of the missing elements that could potentially require clarification.

In this work, we present an implementation of a novel approach to autonomously identify and verbalize such absent information in a knowledge base, in order to enable a robot to actively enhance its knowledge about actions and objects by stepping into dialog with humans.

The contribution of this paper, within the artificial cogni-

tion domain, is the making use of *CNL as mean of language generation*: we use CNL as output language of our doubt verbalization procedure, where doubt is intended as the non autonomously removable uncertainty related to objects involved in the action. Such situation then requires human intervention for appropriate translation to action plans. Our contribution lies in the conceptualization and implementation of the doubt case classification, the discourse abstraction of the robot reply and the verbalization procedure of the latter.

The remainder of this paper presents a description of the adopted technologies (*PRAC, ACE, DRS*); an explanation of how the claims are related to the state of the art; an explanation of the implemented language generation module; a system evaluation and an ending summary comprising results, current limitations and future perspectives.

II. ADOPTED TECHNOLOGIES

Before explaining the details of the dialog-based disambiguation, we now describe the adopted technologies we make use of as source of inferred information (PRAC), and the human-readable target formalization (ACE), used also as support to the verbalization procedure itself (ACE and DRS).

A. Probabilistic Robot Action Cores

Nyga et al. [2] introduced the concept of *Probabilistic Robot Action Cores* (PRAC), which can be thought as abstract, generic event patterns representing sets of inter- and intra- conceptual relations that constitute an abstract event type, assigning an action role to each entity that is affected by the respective action verb. Formally, a PRAC is defined as a conditional probability distribution:

$$P\left(\mathcal{R} \times \mathcal{A} \times \mathcal{C} \mid \sqsubseteq, \preceq\right) \ . \tag{1}$$

$\mathcal{R}$ is the set of all action roles
$\mathcal{A}$ is the set of all action verbs
$\mathcal{C}$ is the set of all class concepts
$\sqsubseteq$ is a taxonomy relation over $\mathcal{C}$
$\preceq$ is a mereological relation over $\mathcal{C}$

For terminology explanations we refer to [2]. As opposed to most approaches towards understanding natural-language instructions, which merely seek to understand what is given by an instruction [5] [6], the PRAC concept is also able to infer what is missing. It combines action-specific and ontological world knowledge in a joint probabilistic first-order representation, which allows to automatically find generalizations from concrete event occurrences at an appropriate level of abstraction. Specifically, PRAC models are represented as a set of action roles (action parameters defining relations among entities involved in an action) and Markov Logic Networks (MLN), a knowledge representation formalism that combines first-order logic and probability theory [7]. Fig. 2 provides an example of the PRAC model for the action core '*flipping*'.

- *Action Core:* Flipping
- *Definition:* An Agent causes a Theme to move with respect to a FixedLocation, generally with a certain Periodicity, without undergoing unbounded translational motion or significant alteration of configuration/shape.
- *Action Roles:*
  - *Theme*: A physical entity that is participating in non-translational motion.
  - *Instrument*: An entity that is used to perform the flipping action.

Fig. 2. "Flipping" Action Core, and an enumeration of its Action Role definitions (partially adopted from *FrameNet* [8]). MLN formulas are not listed for readability.

A PRAC defines a joint probability distribution over the action roles according to Eq. 1, such that arbitrary parameter slots, which are not given in an NL instruction, can be inferred based on what has been stated explicitly in such instruction.

B. Attempto Controlled English & Discourse Representation Structures

Fuchs et al. [9] presented Attempto Controlled English (ACE), a general purpose Controlled Natural Language (CNL), i.e. a subset of standard English with restricted lexicon, syntax and semantics, formally described by a small set of construction rules and a controlled vocabulary. This allows a text in CNL to be read naturally by any person that knows the natural language it stemmed from, even if unaware of the underlying formalizations, and is more readable than other traditional formal languages [10].

Being a formal language, CNL can be proved by automatic theorem proving software, translated into First Order Logic or OWL ontology representations and also be paraphrased into paratactic noun sentences. ACE provides linguistic constructs that are usually present in natural languages, such as countable nouns (e.g. 'robot', 'pancake'), proper names ('John'); universal, existential, generalized quantifiers ('all','a','at least 2'); indefinite pronouns ('somebody'); intransitive, mono- and di-transitive verbs ('sleep', 'like', 'give'); anaphoric references to noun phrases through definite noun phrases and pronouns; composite sentences as compounds of coordination, subordination, quantification, and negation phrases. Fig. 3 provides an example of some of such constructs in an ACE sentence, and also the 'paraphrased understanding', i.e. the breakdown of the latter into paratactic noun phrases, making use of cross-sentence references (i.e. X1, X2 in the example).

ACE: "A robot who does not understand
asks a human that knows."

There is a robot X1. There is a human X2. The human X2 knows. The robot X1 asks the human X2. It is false that the robot X1 understands.

Fig. 3. example sentence in Attempto Controlled English, followed by the 'paraphrased understanding' of such sentence via the use of ACE parser

These paraphrase-obtained noun phrases have a two-way relationship with Discourse Representation Structures (DRS) [11], a format to encode information of multiple sentences, preserving anaphoric references (i.e. *discourse referents*). Fig. 4 provides an example of cross-sentence referencing and universal quantification within the DRS formalism.
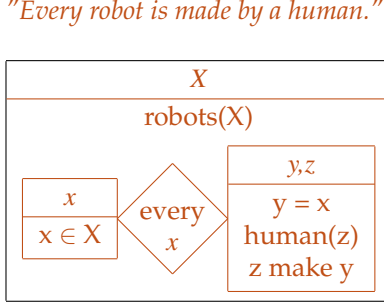

"Every robot is made by a human."

Fig. 4. explanatory ACE sentence & related DRS example describing cross-sentence references and universal quantification

III. RELATED WORK

In reference to filling missing information such as objects or actions in verb-oriented formalizations, a large amount of research has been done in order to provide databases of conceptualizations of actions [8], [12], [13]. However, these projects do not provide computational models for inference and learning, and they do not address the problem of autonomously identifying and closing such gaps of knowledge.

Regarding verbalization, ACE recently has been exploited for uses such as Semantic Web Ontologies [14] and Multilingual semantic wikis [15], while this paper focuses on the contributions of ACE in the artificial cognition domain. Formalisms in artificial cognition oriented towards natural language understanding do exploit grounding of words to abstract objects [16], but do not comprise means of verbal interaction for disambiguation purposes. According to what is known to the authors to date, the ACE verbalization functionality itself [14] has been used uniquely for verbalizing OWL ontologies. While using the same means (i.e. ACE and the DRS verbalizer, the latter being an intermediate phase of the OWL-to-ACE verbalizer) we exploit the system for verbalizing questions and ambiguity statements, in order to verbalize a probabilistic knowledge formalism.

IV. CNL FOR TASK QUERYING

Given a Natural Language Instruction (NLI), the PRAC system caters for action verb and roles understanding, inferring missing candidate objects involved in the action. Unfortunately, this operation can be partially satisfactory and some residual doubt might require explicit verbal clarification, in order to avoid that partially inferred information is translated into action planning.

Fig. 5 represents the interaction that can occur until the robotic assistant reaches a sufficient level of understanding. Taxonomical, temporal, substitution, impossibility clarifications are only some of the *disambiguation cases* in which an explicit verbal task querying is necessary from the robot
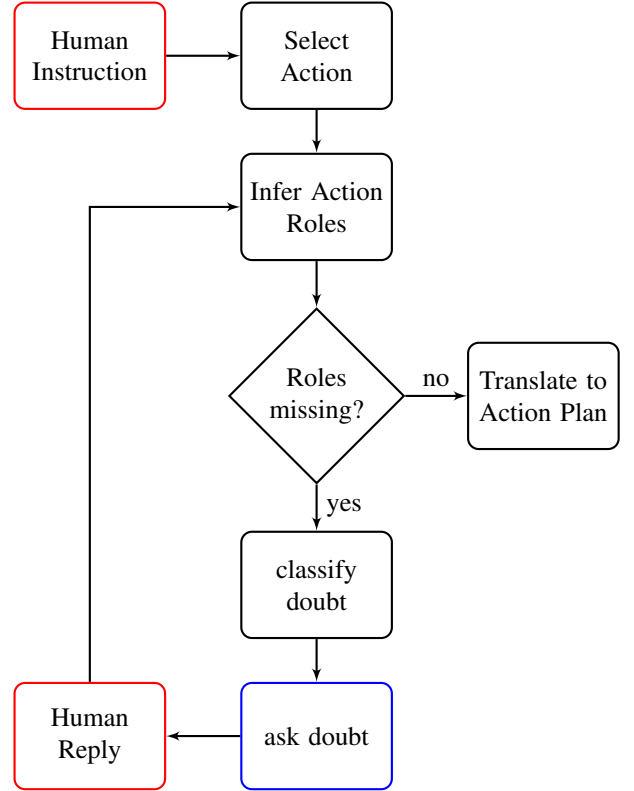

Fig. 5. A high-level flow chart of the dialog-based action role disambiguation procedure.

to the human. The generation of natural-like language is non-trivial given the scalability issues of linguistic factors of the sentence construction (e.g. anaphora resolution, number/gender/particle concordance, subordinate sentence handling, punctuation, verb conjugation). The latter requirements have proven to be fulfilled by Attempto Controlled English (ACE), now used as means of sentence construction. To do so, we make use of the ACE system's verbalization functions, used to generate ACE sentences from ontological knowledge. Specifically for an action-oriented representation, we present our cognitive verbalization procedure, operated for each non-assigned action role:

1) identification of the *disambiguation cases* (i.e. type of doubt)
2) articulation of such doubt in a statement encapsulating what was inferred, and an interrogative sentence
3) integration of the reply, assigning the previously missing action roles

A. Implementation

We now describe the implementation of the system by defining a pseudocode (Algorithm 1) that incorporates references for the sub-functions hereafter described (i.e. C0,C1,C2,C3,C4,C5).

a) action role inference (C0,C1): C0 operates an instantiation of the action template slots with the most likely class concepts, derived from the joint probability distribution of Eq. 1, while C1 retrieves the full enumeration of slots for

Algorithm 1: Action Role Disambiguation via H-R dialog

Data:
- $PRAC$, the joint probability distribution over all used class concepts given ontology knowledge, formally $P\left(\mathcal{R} \times \mathcal{A} \times \mathcal{C} \mid \sqsubseteq, \preceq\right)$.
- $ActionDB$, a template database with all ActionRoles for each ActionVerb

Result: satisfactory assignment of all the ActionRoles required by the template for successful translation into action planning.

begin
    wait for NLI
C0 $K \longleftarrow pracInference(NLI)$
    $t \longleftarrow$ sentence verb from K
C1 $T \longleftarrow retrieveTemplateRoles(t, ActionDB)$
    $U \longleftarrow T \setminus K$
    while $U \neq \emptyset$ **do**
        remove item u from list of U with minimum syntactic relationship arity (known in template)
C2 $doubtCase \longleftarrow doubtCaseIdentification(u)$
C3 $queryType \longleftarrow$ retrieve the grammatical type related to the missing role u
        $sDrsT \longleftarrow pullDrsCase(doubtCase)$
        $qDrsT \longleftarrow pullDrsCase(queryType)$
        $sParam, qParam \longleftarrow$ inferred or known contextual information necessary for the grounding of specific templates of statement and question
        $sDrsGnd, qDrsGnd \longleftarrow$ grounding of $sDrsT$ and $qDrsT$, via syntactic substitution of $sParam$ and $qParam$, respectively.
C4 $aceS \longleftarrow verbalizeDRStoACE(sDrsGnd)$
        $aceQ \longleftarrow verbalizeDRStoACE(qDrsGnd)$
        output aceS
        output aceQ
        wait for $reply$
C5 $N \longleftarrow pracInference(reply)$
        $K \longleftarrow K \cup N$
        $U \longleftarrow T \setminus K$

comparison reasons. A lack of assignment to an action role by C0 can be due to the impossibility of defining a likely candidate (all probability assignments are below a threshold), the presence of manifold candidates (probability assignments are too close), or the optimal candidate is not available in context. For a more formal and in-depth description of such process we refer to [2].

b) case identification (C2): is operated when a role slot stated in our action core template has not yet been assigned for the previously described reasons. Case identification is performed via threshold evaluation of probability values of the most likely concept candidates for the missing role.

More formally, let $fstLikely$ be:

$$\arg\max P(neededRole \mid knownRoles, KB\left(\sqsubseteq, \preceq\right)). \quad (2)$$

We then can describe our selection procedure as:

if $P\left(fstLikely\right) < possibilityThreshold$ **then**
    return "Impossibility"
if $P\left(fstLikely\right) < P\left(sndLikely\right) - proxThreshold$ **then**
    return "Two-choices"
    $\vdots$
return "None"

Such abstract cases are represented in Discourse Representation Structure (DRS) templates that also comprise explanatory information.

TWO-CHOICES:
query_case: doubt between two plausible objects
ACE_template: There is a X1, there is a X2.
drs: drs([A,B],[object(A,X1,countable,na,eq,1)-1/4, object(B,X2,countable,na,eq,1)-1/9])
dependencies: PARAM1-ext, PARAM2-ext; X1, X2

Fig. 6. DRS template of a twofold doubt choice for a role assignment

Fig. 6 provides an example for such template. The explanation of the various fields is the following:

- *query_case* is a high-level descriptive sentence of the case
- *ACE_template* present only for explanatory reasons, is a sentence in ACE that represents, still in a template form, what the output would look like after verbalization
- *drs* is the uninstantiated discourse representation of *ACE_template*: the markers (in the example, X1, X2) will be syntactically substituted with PRAC inferred information upon template grounding
- *dependencies* defines for retrieval and substitution purposes, the type of parameters that are needed to perform grounding, and their corresponding marker in the template. Such parameters have syntactic relationships with other roles involved in the action (described in C3)

c) typed dependency parsing (C3): Together with a statement of the doubt case identification that encapsulates contextual information, a specific object query will also be verbalized in the form of a question. All template action roles have a 2-way relationship with a grammatical type within the scope of the action. These types are abstracted in DRS cases (same as to the DRS modeling described in C2, e.g. in Fig. 8), that need to be retrievable given the missing action role.

The knowledge regarding the association between the action roles and the grammatical type is provided by a *controlled template*, an ACE sentence that comprises all uninstantiated action roles in a possible syntactic configuration, built upon PRAC template model construction (an abstraction

for all instances of that action verb, e.g. 'flipping'). Fig. 7 provides an example of such modeling.
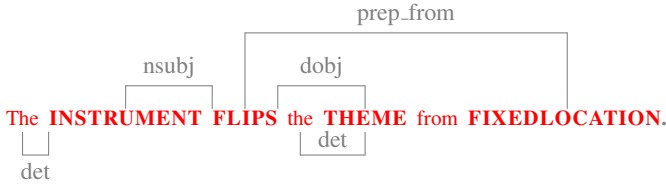

Fig. 7. *Controlled template* example for ActionVerb '*flipping*' (in red), and the typed dependencies among the words of such sentence (in grey)

The grammatical relationships in such a CNL sentence provide a formal understanding of the language-explicit relationships between the entities involved in the action. The relationships and the type of the dependencies from the latter are obtained by processing the sentence with a typed dependency parser (for our implementation, the Stanford Parser [17], for which we also refer to for type clarification).

INSTRUMENTAL:
query_case: preposition of instrument
template: What X1 the X2?
drs: drs([],[question(drs([A,B,C],[query(A,what)-1/1, object(C,X2,countable,na,eq,1)-1/4, predicate(B,X1,A,C)-1/2]))])
dependencies: NSUBJ-left, NSUBJ-right; X1, X2

Fig. 8. DRS template of an 'instrument' object query

d) sentence construction (C4): is to provide a grammatical structure for the grounded discourse abstractions of the case identification statement and the object query question. This is implemented by using the ACE verbalizer functions, providing grounded DRS instances as a formal parameter.

The approach of verbalization of two phrases, namely doubt statement and object question, has been chosen to provide the human with a better understanding of both the uncertainty (e.g. what instrument should be used) and of what has been inferred (e.g. spatula and tongs are the most likely candidates).

e) reply integration (C5): is performed by making use of the previously described action role inference routine on the natural language reply. After retrieving the new action role assignments, we will substitute in the main instance only the newly identified action roles that were previously missing. The pipeline of reasoning is shown with an example in Fig. 9.

V. EVALUATION

As performance measures we take into consideration the natural likeness, the morphosyntactic correctness and the ability to convey the wanted meaning of the CNL output of our verbalizer system.

We operated our evaluation based on two action cores (i.e. Flipping, Filling) that comprised full trained models for
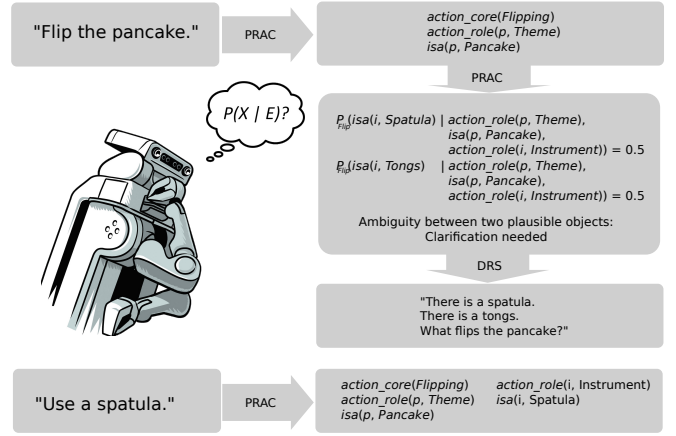

Fig. 9. Reasoning pipeline of a possible instance of disambiguation interaction

PRAC inference in order to verify full pipeline evaluation (example in Fig. 9), and we made use of various other arbitrary NLI sentences to verify correct typed dependency parsing, assignment and grounding of the doubt discourse representations. For the tested domain, the disambiguation verbalization outputs were intelligible and conveyed the meaning, but were not perfectly natural given the lack of the use of modal verbs (i.e. "what flips the pancake?" instead of "what can flip the pancake?"). An evaluated test instance and the related pipeline information is illustrated in Fig. 9.

Given that the verbalizer system is purely a PRAC and DRS based syntactical manipulator, we can assume scalability of our system within the running assumptions and performance bounds of the underlying systems [2] [18].

According to our evaluation, orthography of wording remains intact from NLI to ACE output (the latter partially exploits the same wording), as long as the verbalizer makes use of words that are part of the running ACE system's vocabulary (that can be modified dynamically), otherwise the ACE system will add explicit syntactic tags to highlight the nature of the Part-Of-Speech of such words.

No difference has been observed when making use of action cores based on intransitive, mono- and di-transitive verbs, since the DRS disambiguation cases parameters target typed dependencies that differentiate among direct and indirect objects [17]. Regarding the natural likeness of such sentences, readability studies of ACE have already been undertaken [10].

Regarding the potential ability of conveying the meaning of the doubt given the PRAC abstraction, it is up to who manually constructs the DRS templates, aligned to an ACE output, to be able to exploit the expressiveness of the ACE rules, and will also be constrained by the latter.

VI. OTHER USES OF CNL

Future work can exploit CNL in the artificial cognition context differently, namely by using CNL as serialization of the action oriented formalism, but with the use of semantically unambiguous nouns, therefore with fully deterministic denotations.

Specifically for the formalization of a PRAC model, be it grounded or abstract, we require a format that can serialize an instance or generate the PRAC model template: that will comprise all generative information, namely all roles involved in the action, and data that can create the MLN formulas that evaluate the probability of all possible grammatical types the action roles can be involved in.

We hypothesize that this can be achieved by defining the action roles as nouns in a Controlled Natural Language statement, for human-readability and for preserving grammatical relationships between objects; furthermore we add explicit semantic tags to maintain information regarding semantic disambiguation of objects. An implementation is potentially possible via the use of ACE and semantic tags from WordNet [19]. Fig. 10 provides an example. Any proof of concept of such hypothesis is left as future work.

The AGENT.n.06 FLIPS.v.08 the THEME.n.01 from LOCATION.n.01 with an INSTRUMENT.n.01.

Fig. 10. example of a Controlled Natural Language statement, comprising nouns with semantic tags for action-oriented formalism serialization

VII. Results, Discussions and Conclusions

This paper brings attention to possible uses of Controlled Natural Languages (CNL) in the artificial cognition domain. While already proven as powerful formalism of representation and reasoning for the semantic web [14], our claim is that novel uses of CNL are possible for robotic assistants, specifically as robot-human interface. We have proven via a formalization and a practical implementation that CNL can be exploited as means for sentence construction and target language of verbalization procedures.

However, even if discourse representation is an easier instrument for achieving knowledge engineering, CNL construction is not always straightforward [20]. In fact, the DRS construction of the disambiguation cases has to account for the ACE construction rules (that can present expressiveness limitations) and the asymmetry of what is accepted as correct ACE statement and what can be verbalized (e.g. modals). The verbalizer system, being purely a DRS and PRAC based syntactical manipulator, is constrained by the current implemented features of these and presents similar limitations. This is visible since the verbalization outputs are readable but not perfectly natural-like sentences, and can present scalability issues given by improper PRAC object inference. With the expansion of the expressiveness set of ACE and DRS, future work will aim towards understanding how to make use of such abstractions in order to provide robotic assistants with more language constructs and modalities of speech. Further research will be dedicated to the consolidation of the presented proof of concept, and will focus on the interaction dialogue in order to enable further *learning via human-robot verbal interaction* capabilities.

Acknowledgments

References

[1] J. E. Anderson, "Constraint-directed improvisation for everyday activities," Ph.D. dissertation, University of Manitoba, 1995.

[2] D. Nyga and M. Beetz, "Everything robots always wanted to know about housework (but were afraid to ask)," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vilamoura, Portugal, October, 7–12 2012.

[3] M. Tenorth, D. Nyga, and M. Beetz, "Understanding and executing instructions for everyday manipulation tasks from the world wide web," in *IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, AK, USA, May 3–8 2010, pp. 1486–1491.

[4] M. Tenorth and M. Beetz, "Knowrobknowledge processing for autonomous personal robots," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 4261–4266.

[5] C. Matuszek, D. Fox, and K. Koscher, "Following directions using statistical machine translation," in *Proceedings of the 5th ACM/IEEE international conference on Human-robot interaction*. IEEE Press, 2010, pp. 251–258.

[6] S. Tellex, T. Kollar, S. Dickerson, M. R. Walter, A. G. Banerjee, S. J. Teller, and N. Roy, "Understanding natural language commands for robotic navigation and mobile manipulation." in *AAAI*, 2011.

[7] M. Richardson and P. Domingos, "Markov logic networks," *Machine Learning*, vol. 62, no. 1-2, pp. 107–136, 2006.

[8] C. F. Baker, C. J. Fillmore, and J. B. Lowe, "The berkeley framenet project," in *Proceedings of the 17th international conference on Computational linguistics-Volume 1*. Association for Computational Linguistics, 1998, pp. 86–90.

[9] N. E. Fuchs, K. Kaljurand, and G. Schneider, "Attempto Controlled English Meets the Challenges of Knowledge Representation, Reasoning, Interoperability and User Interfaces," in *FLAIRS 2006*, 2006.

[10] T. Kuhn, "An evaluation framework for controlled natural languages," in *Proceedings of the Workshop on Controlled Natural Language (CNL 2009)*, ser. Lecture Notes in Computer Science, N. E. Fuchs, Ed., vol. 5972. Berlin / Heidelberg, Germany: Springer, 2010, pp. 1–20.

[11] H. Kamp and U. Reyle, *From discourse to logic: Introduction to model theoretic semantics of natural language, formal logic and discourse representation theory*. Kluwer Academic, 1993, vol. 42.

[12] K. K. Schuler, "Verbnet: A broad-coverage, comprehensive verb lexicon," 2005.

[13] P. Kingsbury and M. Palmer, "From treebank to propbank," in *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC-2002)*. Citeseer, 2002, pp. 1989–1993.

[14] K. Kaljurand, "Attempto Controlled English as a Semantic Web Language," Ph.D. dissertation, Faculty of Mathematics and Computer Science, University of Tartu, 2007.

[15] K. Kaljurand and T. Kuhn, "A multilingual semantic wiki based on Attempto Controlled English and Grammatical Framework," in *Proceedings of the 10th Extended Semantic Web Conference (ESWC 2013)*. Springer, 2013.

[16] S. Lemaignan, R. Ros, E. A. Sisbot, R. Alami, and M. Beetz, "Grounding the interaction: Anchoring situated discourse in everyday human-robot interaction," *International Journal of Social Robotics*, vol. 4, no. 2, pp. 181–199, 2012.

[17] M.-C. de Marneffe and C. D. Manning, "The stanford typed dependencies representation," in *COLING Workshop on Cross-framework and Cross-domain Parser Evaluation*, 2008. [Online]. Available: pubs/dependencies-coling08.pdf

[18] N. E. Fuchs, K. Kaljurand, and T. Kuhn, "Discourse Representation Structures for ACE 6.6," Department of Informatics, University of Zurich, Zurich, Switzerland, Tech. Rep. ifi-2010.0010, 2010.

[19] G. A. Miller, "Wordnet: A lexical database for english," *Communications of the ACM*, vol. 38, pp. 39–41, 1995.

[20] R. Schwitter, "A layered controlled natural language for knowledge representation," in *Machine Translation, Controlled Languages and Specialised Languages: Special Issue of Linguisticae Investigationes*, 2005, pp. 85–106.