Structured Learning for Context-aware Spoken Language Understanding of Robotic Commands

Andrea Vanzo[†] and **Danilo Croce**[‡] and **Roberto Basili**[‡] and **Daniele Nardi**[†]
[†]Sapienza University of Rome
Dept. of Computer, Control and Management Engineering "Antonio Ruberti"
[‡]University of Roma, Tor Vergata
Dept. of Enterprise Engineering
{vanzo,nardi}@dis.uniroma1.it,{croce,basili}@info.uniroma2.it

Abstract

Service robots are expected to operate in specific environments, where the presence of humans plays a key role. A major feature of such robotics platforms is thus the ability to react to spoken commands. This requires the understanding of the user utterance with an accuracy able to trigger the robot reaction. Such correct interpretation of linguistic exchanges depends on physical, cognitive and language-dependent aspects related to the environment. In this work, we present the empirical evaluation of an adaptive Spoken Language Understanding chain for robotic commands, that explicitly depends on the operational environment during both the learning and recognition stages. The effectiveness of such a context-sensitive command interpretation is tested against an extension of an already existing corpus of commands, that introduced explicit perceptual knowledge: this enabled deeper measures proving that more accurate disambiguation capabilities can be actually obtained.

1 Introduction

In recent years, one of the most challenging issues that Service Robotics is facing is the automation of high level and collaborative interactions between humans and robots. In such a robotic context, human language is the most natural way of communication as for its expressiveness and flexibility. However, an effective communication in natural language between humans and robots is challenging mostly for the different cognitive abilities it involves. For a robot to react to a simple command like "*take the mug in the kitchen*", a number of implicit assumptions should be met. First, at least two entities, a `mug` and a `kitchen`, must exist in the environment and the speaker must be aware of such entities. Accordingly, the robot must have access to an inner representation of its world, e.g., an explicit map of the environment. Second, mappings from lexical references to real world entities must be developed or made available. In this respect, the *Grounding* process (Harnad, 1990) links symbols (e.g., words) to the corresponding perceptual information. Hence, robot interactions need to be *grounded*, as meaning depends on the state of the physical world and the interpretation crucially interplays with perception, as pointed out by psycho-linguistic theories (Tanenhaus et al., 1995). The integration of perceptual information derived from the robot's sensors with an ontologically motivated description of the world has been adopted as an augmented representation of the environment, in the so-called *semantic maps* (Nüchter and Hertzberg, 2008). In these maps, the existence of real world objects can be associated to *lexical* information, in the form of entity names given by a knowledge engineer or spoken by a user for a pointed object, as in Human-Augmented Mapping (Diosi et al., 2005; Gemignani et al., 2016). While Command Interpretation for Interactive Robotics has been mostly carried out over the only evidence specific to the linguistic level (see, for example, (Chen and Mooney, 2011; Matuszek et al., 2012)), we argue that a proper Spoken Language Understanding (SLU) for Human-Robot Interaction should be context-aware, in the sense that both the user and the robot live in and make references to a shared environment. For example, in the above command, "*taking*" is the intended action whenever a mug is actually in the kitchen, so that "*the mug in the kitchen*" refers to a single argument. On the contrary, the command may refer to a "*bringing*" action, when no mug is in the kitchen and *the mug* and *in the kitchen*

Proceedings of the First Workshop on Language Grounding for Robotics, pages 25–34,
Vancouver, Canada, July 30 - August 4, 2017.

correspond to different semantic roles. We are interested in an approach for the interpretation of robotic spoken commands that is consistent with (*i*) the world (with all the entities composing it), (*ii*) the Robotic Platform (with its inner representations and capabilities), and (*iii*) the linguistic information derived from the user's utterance.

In this paper, we foster machine learning methodologies for Spoken Language Understanding that force the above research perspective: this is obtained by extending the linguistic evidence that can be extracted from the uttered commands with perceptual evidence directly derived by the semantic map of a robot. In particular, the interpretation process is modeled as a sequence labeling problem where the final labeler is trained by applying Structured Learning methods over realistic commands expressed in domestic environments, as in (Bastianelli et al., 2017). The resulting interpretations adhere to Frame Semantics (Fillmore, 1985): this well-established theory provides a strong linguistic foundations to the overall process while enforcing its applicability, as it is made independent of the vast plethora of existing robotic platforms. Such methodologies have been implemented in a free and ready-to-use framework, here presented, whose name is *LU4R* - an adaptive spoken **L**anguage **U**nderstanding framework for(**4**) **R**obots. LU4R is entirely coded in Java and, thanks to its Client/Server architectural design, it is completely decoupled from the robot, enabling for an easy and fast deployment on every platform[1].

As the aforementioned approaches rely on realistic data, in this work we also present an extended version of *HuRIC* - a **Hu**man **R**obot **I**nteraction **C**orpus, originally introduced in (Bastianelli et al., 2014) This resource is a collection of realistic spoken commands that users might express towards generic service robots. In this resource, each sentence is labeled with morpho-syntactic information (e.g., dependency relations, POS tags, . . .), along with its correct interpretation in terms of semantic frames (Baker et al., 1998). In our extension, each annotated sentence is paired with a semantic representation of the world, that justifies the command itself. To the best of our knowledge this is the first corpus providing such a rich representation of a robotic spoken command[2].

This extension of HuRIC supports a broader evaluation of LU4R chain against the information introduced by perceptual knowledge. We observed a significant increase in performance w.r.t. inherent ambiguities of the language, whose outcomes are encouraging for the deployment of such system in realistic applications.

The rest of the paper is structured as follows. Section 2 provides a short survey of existing approaches to SLU for Human-Robot Interaction. Section 3 describes the semantic analysis process that represents the core of LU4R. In Section 4, an architectural description of the entire framework is provided, as well as an overall introduction about its integration with a generic robot. Section 5 describes the extension of HuRIC, while in Section 6 we provide empirical evidence demonstrating the applicability of the proposed system in the interpretation of robotic commands, by reporting our experimental results. In Section 7 we draw some conclusions.

2 Related Work

In Robotics, some solutions for the interpretation of spoken commands have been modeled using grammar-based approaches. In general, they provide mechanisms to enrich the syntactic structure with semantic information, to build a semantic representation during the transcription process (Bos, 2002; Bos and Oka, 2007).

Other approaches are based on formal languages, as in (Kruijff et al., 2007; Thomason et al., 2015), where Combinatory Categorial Grammar (CCG) are applied for spoken dialogues in Human-Robot Interaction, and in (Perera and Veloso, 2015) where template-based algorithms allow extracting semantic interpretations of robotic commands by applying specific templates over the corresponding syntactic trees.

Data-driven methods have been also applied to command interpretation for robotic applications. Examples are (MacMahon et al., 2006) and (Chen and Mooney, 2011), where the parsing of route instructions is addressed as a Statistical Machine Translation task between the human language and a synthesized robot language. The same approach is applied in (Matuszek et al., 2010) to learn translation models between natural language and formal descriptions of paths. A probabilistic CCG is used in (Matuszek et al., 2012) to map natu-

[1] LU4R can be downloaded at `http://sag.art.uniroma2.it/lu4r.html`

[2] The extended version of HuRIC will be released at `http://sag.art.uniroma2.it/huric.html`

ral navigational instructions into robot executable commands. The same problem is faced in (Kollar et al., 2010; Duvallet et al., 2013), where Spatial Description Clauses are parsed from sentences through sequence labeling approaches. In (Tellex et al., 2011), the authors address natural language instructions about motion and grasping, that are mapped into Generalized Grounding Graphs (G^3). In (Fasola and Mataric, 2013a,b), Spoken Language Understanding (SLU) for pick-and-place instructions is performed through a Bayesian classifier trained over a specific corpus. In (Misra et al., 2016), the authors define a probabilistic approach to ground natural language instructions within a changing environment.

In this paper we present a data-driven approach that integrates an explicit semantic representation with linguistic generalization induced through machine learning. On the one hand, the interpretation is carried out according to the Frame Semantics paradigm (Fillmore, 1985), thus resulting in a principled meaning representation formalism. Moreover, a context-dependent interpretation process is realized: knowledge derived from perceptual evidence is made available and directly used to discriminate against conflicting interpretations. Perceptual information is here represented through an ontologically motivated description of the surrounding environment, i.e., a *semantic map* (Nüchter and Hertzberg, 2008). The semantic map is an explicit representation of the knowledge about surroundings, acquired to enable reasoning over environments, objects and properties. In the map, the existence and position of real world objects is associated to lexical information, in the form of entity class names. On the other hand, machine learning depends on such perceptual information, thus inducing the contextual preconditions of the involved disambiguation choices from real examples, i.e. sentence-map pairs. The process can thus provide different interpretations of one sentence against different maps and realizes a highly reusable and mostly domain-independent model of grounded interpretation.

3 The Language Understanding Cascade

A command interpretation system for a robotic platform must produce interpretations of user utterances. In this paper, we consider Frame Semantics (Fillmore, 1985), the formalization promoted in the FrameNet (Baker et al., 1998) project, where actions expressed in user utterances can be modeled as *semantic frames*. Each frame represents a micro-theory about a real world situation, e.g., the actions of *bringing*, *motion* or *manipulation*. Such micro-theories encode all the relevant information needed for their correct interpretation. This information is represented in FrameNet via the so-called *frame elements*, whose role is to specify the participating entities in a frame, e.g., the THEME frame element represents the object that is taken in a *bringing* action.

As an example, let us consider the sentence: "*take the pillow to the couch*". This sentence can be intended as a command whose effect is to instruct a robot that, in order to achieve the task, has to: *(i)* move towards a pillow, *(ii)* pick it up, *(iii)* move to the couch and, finally, *(iv)* release the object on the couch. The language understanding cascade should produce its FrameNet-annotated version:

$$[take]_{Bringing}[the\ pillow]_{\text{THEME}}[to\ the\ couch]_{\text{GOAL}} \quad (1)$$

Semantic frames can thus provide a cognitively sound bridge between the actions expressed in the language and the implementation of such actions in the robot world, namely plans and operations.

The whole SLU process has been designed as a cascade of reusable components, as shown in Figure 1. As we deal with vocal commands, their (possibly multiple) hypothesized transcriptions derived from an Automatic Speech Recognition (ASR) engine constitute the input of this process. It is composed by four modules, whose final output is the interpretation of an utterance, to be used to implement the corresponding robotic actions. First, **Morpho-syntactic analysis** is performed over the available utterance transcriptions by applying morphological analysis, Part-of-Speech tagging and syntactic analysis. In particular, dependency trees are extracted from the sentence as well as POS tags, as shown in Figure 2. Then, if more than one transcription hypothesis is available, the **Re-ranking** module can be activated to compute a new ranking of the hypotheses, in order to get the best transcription out of the initial ranking. This module is realized through a learn-to-rank approach, where a Support Vector Machine exploiting a combination of linguistic kernels is applied, according to (Basili et al., 2013). Third, the best transcription is the input of the **Action Detection** (AD) component. The evoked frames in a sentence are detected, along with the

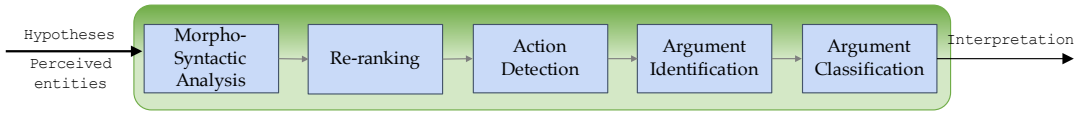

Figure 1: The SLU cascade

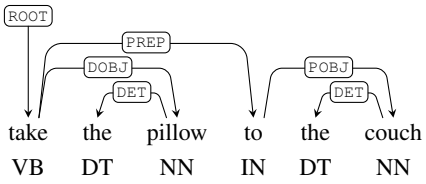

Figure 2: Example of a dependency graph associated to "*take the pillow to the couch*"

corresponding evoking words, the so-called lexical units. Let us consider the recurring sentence: the AD should produce the following interpretation $[take]_{Bringing}$ *the pillow to the couch*. The final step is the **Argument Labeling**, where a set of frame elements is retrieved for each frame. This process is realized in two sub-steps. First, the *Argument Identification* (AI) finds the spans of all the possible frame elements, producing the following form $[take]_{Bringing}$ $[the\ pillow]$ $[to\ the\ couch]$. Then, the *Argument Classification* (AC) assigns the suitable label (i.e., the frame element) to each span thus returning the final tagging shown in the Example (1).

The AD, AI and AC steps are modeled as a sequence labeling task, as in (Bastianelli et al., 2016). The Markovian formulation of a structured SVM proposed in (Altun et al., 2003) is applied to implement the labeler, known as SVM^{hmm}. In general, this learning algorithm combines a local discriminative model, which estimates the individual observation probabilities of a sequence, with a global generative approach to retrieve the most likely sequence, i.e., tags that better explain the whole sequence. In other words, given an input sequence $\mathbf{x} = (x_1 \dots x_l) \in \mathcal{X}$ of feature vectors $x_1 \dots x_l$, SVM^{hmm} learns a model isomorphic to a k-order Hidden Markov Model, to associate $\mathbf{x}$ with a set of labels $\mathbf{y} = (y_1 \dots y_l) \in \mathcal{Y}$.

A sentence s is here intended as a sequence of words w_i, each modeled through a feature vector x_i and associated to a dedicated label y_i, specifically designed for each interpretation process[3]: in any case, features combine linguistic evidence from a targeted sentences, but also properties derived from the semantic map (when available) in order to synthesize information about existence and position of entities around the robot, as discussed in more details in (Bastianelli et al., 2016). During training, the SVM algorithm associates words to step-specific labels: linear kernel functions are applied to different types of features, ranging from linguistic to perception-based features, and linear combinations of kernels are used to integrate independent properties. At classification time, given a sentence $s = (w_1 \dots w_{|s|})$, the SVM^{hmm} efficiently predicts the tag sequence $\mathbf{y} = (y_1 \dots y_{|s|})$ using a Viterbi-like decoding algorithm. More details about the construction of feature vectors x_i are reported in (Bastianelli et al., 2016).

Notice that both the re-ranking and the semantic parsing phases can be realized in two different settings, depending on the type of features adopted in the labeling process. It is thus possible to rely upon linguistic information to solve the given task, or also on perceptual knowledge coming from a semantic map. In the first case, that we call *basic* setting, the information used to solve the task comes from linguistic inputs, as the sentence itself or external linguistic resources. These models correspond to the methods discussed in (Bastianelli et al., 2017; Basili et al., 2013). In the second case, the *simple* setting, when perceptual information is made available to the chain, a context-aware interpretation is triggered, as in (Bastianelli et al., 2016). Such perceptual knowledge is mainly exploited through a *linguistic grounding* mechanism. This lexically-driven grounding is estimated through distances between filler (i.e., argument heads) and entity names. Such a semantic distance integrates metrics over word vectors descriptions and phonetic similarity. Word semantic vectors are here acquired through corpus analysis, as in Distributional Lexical Semantic paradigms (Turney and Pantel, 2010). They allow to map referential elements, such as lexical fillers, e.g., *couch*, to entities, e.g., a `sofa`, by thus modeling synonymy or co-hyponymy. Conversely, phonetic similarities

[3] More details about the labeling notation can be found in (Bastianelli et al., 2016)

are smoothing factors against possible ASR transcription errors, e.g., *pitcher* and *picture*: this allows to actually cope with the noisy phenomena characterizing spoken language.

Once links between fillers and entities have been activated, they act as abductive hypothesis: they inspire features related to individual words that express perceptual information (e.g. presence/absence of referred objects in the environment or spatial relations between them) as well as lexical knowledge (e.g. semantic and phonetic similarity between entity names and uttered references). The labeler trained over such richer descriptions is made thus sensitive to perceptual information both in the learning and the tagging process. As a side effect, the above mechanism provides the robot with the set of linguistically-motivated groundings, that can be potentially used for any further grounding process.

This information can be crucial in the correct interpretation of ambiguous commands, which depends on the specific environmental setting the robot is operating into. A clear example is the command *"bring the pillow on the couch in the living room"*. Such a sentence may have two different interpretations, according to the configuration of the environment. In fact, when the couch is located into the living room, the goal of the *Bringing* action is the couch and interpretation will be: $[bring]_{Bringing}[the\ pillow]_{\text{THEME}}[on\ the\ couch\ in\ the\ living\ room]_{\text{GOAL}}$. Conversely, if the couch is outside the living room, it means that probably the pillow is already on the couch. Hence, the interpretation of the sentence will be different, due to different argument spans, and the couch becomes the goal of the *Bringing* action: $[bring]_{Bringing}[the\ pillow\ on\ the\ couch]_{\text{THEME}}[in\ the\ living\ room]_{\text{GOAL}}$.

Additional details about the pure linguistic approach can be found in (Bastianelli et al., 2017).

4 The LU4R Framework

The architecture of the system considers two main actors, as shown in Figure 3: the *Robotic Platform* and *LU4R*, where the processing cascade of the latter component have been introduced in the previous Section.

The Client-Server communication schema between LU4R and the Robot allows for the independence from the Robotic Platform, in order to maximize the re-usability and integration in heterogeneous robotic settings. LU4R exhibits semantic capabilities (e.g., disambiguation, predicate detection or grounding into robotic actions and environments) that are designed to be general enough to be representative of a large set of application scenarios.

It is obvious that an interpretation process must be achieved even when no information about the domain/environment is available, i.e., a scenario involving a *blind* but speaking robot, or when the actions a robot can perform are not made explicit. At the same time, the proposed SLU cascade makes available methods to specialize its semantic interpretation process to individual situations where more information is available about goals, the environment and the robot capabilities. These methods are expected to support the optimization of the core SLU process against a specific interactive robotics setting, in a cost-effective manner. In fact, whenever more information about the environment perceived by the robot (e.g., a semantic map) or about its capabilities is provided, the interpretation of a command can be improved by exploiting a more focused scope.

In order to better understand the different operating modalities of LU4R, some assumptions toward the Robotic Platform must be made explicit: this will allow to precisely establish functionalities and resources that the robot needs to provide to unlock the more complex processes. These information will be used to express the experience that the robot is able to share with the user (i.e., the perceptual knowledge about the environment where the linguistic communication occurs and some lexical information and properties about objects in the environment) and some level of awareness about its own capabilities (e.g., the primitive actions that the robot is able to perform, given its hardware components).

4.1 The Robotic Platform

The overall framework contemplates a generic Robotic Platform, whose task, domain and physical setting are not necessarily specified. In order to make the SLU process independent of the above specific aspects, we assume that the platform requires, at least, the following modules: *(i)* an Automatic Speech Recognition (ASR) system, *(ii)* a SLU Orchestrator, *(iii)* a Grounding and Command Execution Engine, and *(iv)* a Physical Robot. The ASR component currently re-

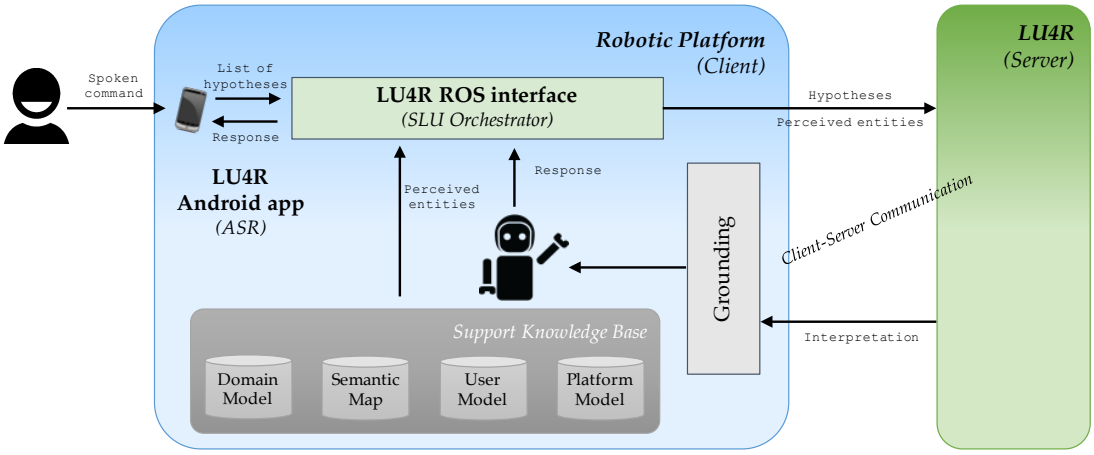

Figure 3: The architecture of the LU4R framework

Number of examples	656
Number of frames	18
Number of predicates	767
Number of roles	34
Predicates per sentence	1.17
Sentences per frame	36.44
Roles per sentence	2.04
Entities per sentence	7.29

Table 1: Some statistics of the corpus

Motion	143	*Bringing*	153
Cotheme	39	*Locating*	90
Inspecting	29	*Taking*	80
Change_direction	11	*Arriving*	12
Giving	10	*Placing*	52
Closure	19	*Change_operat_state*	49
Being_located	38	*Attaching*	11
Releasing	9	*Perception_active*	6
Being_in_category	11	*Manipulation*	5

Table 2: Distribution of frames over the corpus

alized exploits the *LU4R Android app* whereas the SLU orchestrator is implemented as a ROS node, through the *LU4R ROS interface*. Additionally, the optional *Support Knowledge Base* component is expected to interface the different involved knowledge sources and support their maintenance: this provides the contextual information discussed above.

5 A Perceptual Corpus of Robotic Commands

The computational paradigms adopted here are based on machine learning techniques and depend strictly on the availability of training data. In order to train and test our framework, a proper resource that collects both linguistic and perceptual information is required. To this end, we extended the Human-Robot Interaction Corpus[4] (HuRIC), formerly presented in (Bastianelli et al., 2014), by pairing each English sentence with the corresponding perceptual evidence that justifies the targeted semantics.

HuRIC is based on Frame Semantics and captures cognitive information about situations and events expressed in sentences. The corpus does not include system or robot-dependent sentences or formalisms. Instead, it contains information strictly related to Natural Language Semantics, decoupled from specific tasks. The corpus exploits different situations representing possible commands given to a robot in a house environment. Each sentence is paired with a set of audio files representing robot commands and its corresponding correct transcription. Each sentence is then annotated with: lemmas, POS tags, dependency trees and Frame Semantics. Semantic frames and frame elements are used to represent the meaning of commands, as they reflect the actions a robot can accomplish in a home environment. In this respect, the AMR representation of the Example 1 is

```
(t1   / take-Bringing
      :Theme (b1 / pillow)
      :Goal (t2 / couch)
)
```

In this way, HuRIC can potentially be used to train all the modules of the processing chain presented in Section 4.

With respect to the previous release, we extended HuRIC by pairing each sentence with the

[4] Available at http://sag.art.uniroma2.it/huric. The download page also contains a detailed description of the release format.

corresponding semantic map, composed of all entities populating the environment and presumably "perceived" by the robot. Each entity is represented by the following set of information.

The **Atom** is a unique identifier of the entity, whereas the **Type** of each entity, reflects the class to which each specific entity belongs[5].

The **Preferred Lexical Reference** is used to refer to a class of objects; it is crucial in order to enable the grounding between the commands uttered by the user and the entities within the environment. For example, an entity of the class `table` can be referred by the word *desk*.

Finally, the position of each entity is essential to determine shallow spatial relations between entities, e.g., whether two objects are *near* or *far* from each other. To this end, each entity is associated with its **Coordinate** in the world, in terms of planar coordinates (x, y), elevation (z) and *angle* as the orientation. We adopted a simple numerical scaling that discretized the map.

Table 1 shows the number of annotated sentences, number of frames, along with the average number of entities per sentence. Each entity involved in the command, e.g., *mug* and *kitchen* in the Example 1, is provided with one lexical reference, not necessarily the same word used in the command (e.g. using a synonym such as *cushion* or *sofa*). Detailed statistics about the number of sentences for each frame are reported in Table 2.

6 Experimental Evaluation

In order to provide evidence about the benefits of perceptual knowledge, we report an evaluation of the interpretation process of robotic commands over the enhanced version of HuRIC, i.e., contemplating the semantic maps for each sentence.

Table 3 shows the results obtained. The results, expressed in terms of *Precision*, *Recall* and *F1 measure*, focus on the semantic interpretation process, in particular Action Detection (AD), Argument Identification (AI) and Argument Classification (AC) steps, addressing two possible configurations: a basic setting where only linguistic information is exploited (i.e., *noSM*, as the semantic maps are ignored), and the configuration where semantic maps are included into the learning loop (i.e., *SM*). F1 scores measure the quality of a specific module. While in the AD step the F1 refers

[5]Notice that an entity can be an object, e.g., *couch*, *pillow*, or a location, e.g., *bedroom*

	Precision	***Recall***	***F1-Measure***
		AD	
noSM	94.73 ± 1.21	94.02 ± 1.51	94.37 ± 1.00
SM	95.69 ± 1.40	96.90 ± 1.90	$\mathbf{96.29} \pm 1.56$
		AI	
noSM	88.95 ± 2.24	88.22 ± 2.08	88.57 ± 1.65
SM	91.34 ± 1.73	91.72 ± 1.14	$\mathbf{91.53} \pm 1.43$
		AC	
noSM	93.05 ± 1.05	93.05 ± 1.05	93.05 ± 1.05
SM	94.02 ± 1.25	94.02 ± 1.25	$\mathbf{94.02} \pm 1.25$

Table 3: Experimental evaluation of the semantic interpretation process

to the ability to extract the correct frame(s) (i.e., robot action(s) expressed by the user) evoked by a sentence, in the AI step it evaluates to the correctness of the predicted argument spans. Finally, in the AC step the F1 measures the accuracy of the classification of individual arguments. The experiments have been performed in a 5-fold cross validation setting. In this respect, Table 3 provides also the *standard deviations* among the different folds. We tested each sub-module in isolation, feeding each step with gold information provided by the previous step in the chain. Moreover, the evaluation has been carried out considering the correct transcriptions, i.e., not contemplating the error introduced by the Automatic Speech Recognition system.

The overall results are encouraging for the application of the proposed approach in realistic scenarios. In fact, the F1 is always higher than 94% in the recognition of semantic predicates used to express intended actions (AD). The system is able to recognize the involved entities (AC) with high accuracy as well, with a F1 higher than 93% in both *noSM* and *SM* settings. This result is surprising when analyzing the complexity of the task. In fact, the classifier is able to cope with a high level of uncertainty, as the amount of possible semantic roles is sizable, i.e., 34. In general, the most challenging task seems to be the ability to recognize the spans composing a single frame element (AI).

Regarding the *noSM* setting, i.e., only linguistic information, one of the most frequent error concerns the ambiguity of the *"take"* verb. In fact, as explained in the previous sections, the interpretation of such verb may be different (i.e., either *Bringing* or *Taking*), depending on the configuration of the environment. As this particular setting does not provide any kind of perceptual information, the system is not able to correctly discrimi-

nate among them. Hence, the resulting interpretation will be wrong, as it does not reflect the semantics that is motivated by the environment. In terms of F1 measure, this issue affects mainly the Argument Identification step (AI), rather than the Action Detection (AD) one, as for each (possibly) wrong frame, there could be more than two (possibly) wrong arguments. For example, the sentence *"take the mug in the kitchen"* will be probably recognized to be a *Taking* action, even though it is labeled as *Bringing*, i.e., *mug* and *kitchen* are supposed to be far in the environment. While the AD step will receive just one penalty for the wrong recognized action, the AI step is penalized twice, as two arguments were expected by the gold standard annotation, i.e., the *the mug* as THEME and the *in the kitchen* as GOAL, instead of one, i.e., *the mug in the kitchen* as a single THEME argument.

When looking at the *SM* setting, it seems that the injection of perceptual knowledge into the semantic analysis process is able to mitigate the effect of the aforementioned phenomena and each SLU step gains in predictive performance. In the case of AD, the information about the entities shows a relative improvement of $+2.03\%$ in terms of F1 (94.37% vs 96.29%). This means that the semantic map allows to predict the intended action more accurately, whenever the underlying semantic ambiguity depends on the configuration of the environment. The tight correlation between the predicted action and the frame elements suggests a similar behavior in Argument Identification. In fact, as well as for the AD, in the AI step perceptual knowledge reveal its support in predicting the correct spans of semantic arguments, with a relative improvement of $+3.34\%$ w.r.t. the F1 score. Though a lower gain is observed ($+1.04\%$), the introduction of Distributional Semantics improves the ability of recognizing the correct frame element for a given argument span, i.e., AC step. This is probably due to the lexical generalization provided by the word embeddings, whenever alternative naming are used to refer to an entity of the semantic map.

Finally, small values of standard deviation suggest that the system seems to be rather stable across the different iterations of the experiment and that the results do not depend on specific splits of the entire dataset.

7 Conclusions

In this paper, we presented a comprehensive framework for the design of robust natural language interfaces for Human-Robot Interaction (HRI). The corresponding implementation is specifically designed for the automatic interpretation of spoken commands in domestic environments. The proposed solution relies on Frame Semantics and supports a structured learning approach to language processing able to map individual sentence transcriptions to meaningful commands. A hybrid discriminative and generative learning method is proposed to map the interpretation process into a cascade of sentence annotation tasks. The interpretation of commands is made dependent on the robot's environment; in fact the adopted training annotations not only express linguistic evidence from source utterances, but also account for specific perceptual information derived from a reference map. In this way the semantic map aspects useful to interpretation are expressed via feature modeling with the structured learning mechanism applied. Such perceptual knowledge is derived from a semantically-enriched implementation of a robot map, i.e., its semantic map. It expresses information about the existence and position of entities surrounding the robot: as this is also available to the user, this information is crucial to disambiguate predicates and role assignments.

To this end, we trained the machine learning processes by using an extended version of *HuRIC*, the Human Robot Interaction Corpus. This corpus, originally composed by sentences in English, now benefits from the introduction of such semantic maps, expressed as lists of entities and supporting the research in natural language interfaces for Robots in such language. The empirical results obtained over the perceptual version of the dataset show a significant improvement w.r.t. the pure linguistic process. This confirms the effectiveness of the proposed processing chain.

Future research will also focus on the extension of the proposed methodology, e.g., by considering spatial relations between entities in the environment or their physical characteristics, such as their color and the application of this solution in interactive question answering or dialogue with robots.

References

Yasemin Altun, I. Tsochantaridis, and T. Hofmann. 2003. Hidden Markov support vector machines. In *Proc. of ICML*.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The berkeley framenet project. In *Proceedings of ACL and COLING*. pages 86–90.

Roberto Basili, Emanuele Bastianelli, Giuseppe Castellucci, Daniele Nardi, and Vittorio Perera. 2013. Kernel-based discriminative re-ranking for spoken command understanding in hri. In *AI* IA 2013: Advances in Artificial Intelligence*, Springer International, pages 169–180.

Emanuele Bastianelli, Giuseppe Castellucci, Danilo Croce, Roberto Basili, and Daniele Nardi. 2014. Huric: a human robot interaction corpus. In *Proc. of LREC 2014*. Reykjavik, Iceland.

Emanuele Bastianelli, Giuseppe Castellucci, Danilo Croce, Roberto Basili, and Daniele Nardi. 2017. Structured learning for spoken language understanding in human-robot interaction. *The International Journal of Robotics Research* 0(0):To appear in. https://doi.org/10.1177/0278364917691112.

Emanuele Bastianelli, Danilo Croce, Andrea Vanzo, Roberto Basili, and Daniele Nardi. 2016. A discriminative approach to grounded spoken language understanding in interactive robotics. In *Proc. of the 25th IJCAI, New York*.

Johan Bos. 2002. Compilation of unification grammars with compositional semantics to speech recognition packages. In *Proceedings of the 19th International Conference on Computational Linguistics - Volume 1*. Association for Computational Linguistics, Stroudsburg, PA, USA, COLING '02, pages 1–7. https://doi.org/10.3115/1072228.1072323.

Johan Bos and Tetsushi Oka. 2007. A spoken language interface with a mobile robot. *Artificial Life and Robotics* 11(1):42–47.

David L. Chen and Raymond J. Mooney. 2011. Learning to interpret natural language navigation instructions from observations. In *Proc. of the 25th AAAI Conference*. pages 859–865.

Albert Diosi, Geoffrey R. Taylor, and Lindsay Kleeman. 2005. Interactive SLAM using laser and advanced sonar. In *Proc. of the 2005 International Conference on Robotics and Automation*. pages 1103–1108.

Felix Duvallet, Thomas Kollar, and Anthony Stentz. 2013. Imitation learning for natural language direction following through unknown environments. In *2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, May 6-10, 2013*. pages 1047–1053. https://doi.org/10.1109/ICRA.2013.6630702.

Juan Fasola and Maja J Mataric. 2013a. Using semantic fields to model dynamic spatial relations in a robot architecture for natural language instruction of service robots. In *International Conference on Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ*. pages 143–150.

Juan Fasola and Maja J. Mataric. 2013b. Using spatial semantic and pragmatic fields to interpret natural language pick-and-place instructions for a mobile service robot. In *Social Robotics: 5th International Conference, ICSR 2013, Bristol, UK, October 27-29, 2013, Proceedings*, Springer International Publishing, pages 501–510.

Charles J. Fillmore. 1985. Frames and the semantics of understanding. *Quaderni di Semantica* 6(2):222–254.

Guglielmo Gemignani, Roberto Capobianco, Emanuele Bastianelli, Domenico Daniele Bloisi, Luca Iocchi, and Daniele Nardi. 2016. Living with robots. *Robot. Auton. Syst.* 78(C):1–16. https://doi.org/10.1016/j.robot.2015.11.001.

S. Harnad. 1990. The symbol grounding problem. *Physica D: Nonlinear Phenomena* 42(1-3):335–346.

Thomas Kollar, Stefanie Tellex, Deb Roy, and Nicholas Roy. 2010. Toward understanding natural language directions. In *Proceedings of the 5th ACM/IEEE*. Piscataway, NJ, USA, HRI '10, pages 259–266.

Geert-Jan M. Kruijff, H. Zender, P. Jensfelt, and Henrik I. Christensen. 2007. Situated dialogue and spatial organization: What, where... and why? *International Journal of Advanced Robotic Systems* 4(2).

Matt MacMahon, Brian Stankiewicz, and Benjamin Kuipers. 2006. Walk the talk: connecting language, knowledge, and action in route instructions. In *proceedings of the 21st national conference on Artificial intelligence - Volume 2*. AAAI Press, AAAI'06, pages 1475–1482.

Cynthia Matuszek, Dieter Fox, and Karl Koscher. 2010. Following directions using statistical machine translation. In *Proceedings of the 5th ACM/IEEE International Conference on Human-robot Interaction*. IEEE Press, Piscataway, NJ, USA, HRI '10, pages 251–258.

Cynthia Matuszek, Evan Herbst, Luke S. Zettlemoyer, and Dieter Fox. 2012. Learning to parse natural language commands to a robot control system. In Jaydev P. Desai, Gregory Dudek, Oussama Khatib, and Vijay Kumar, editors, *ISER*. Springer, volume 88 of *Springer Tracts in Advanced Robotics*, pages 403–415.

Dipendra K. Misra, Jaeyong Sung, Kevin Lee, and Ashutosh Saxena. 2016. Tell me dave: Context-sensitive grounding of natural language to manipulation instructions. *The International Journal of Robotics Research* 35(1-3):281–300. https://doi.org/10.1177/0278364915602060.

Andreas Nüchter and Joachim Hertzberg. 2008. Towards semantic maps for mobile robots. *Robot. Auton. Syst.* 56(11):915–926.

Vittorio Perera and Manuela M. Veloso. 2015. Handling complex commands as service robot task requests. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*. pages 1177–1183.

M. Tanenhaus, M. Spivey-Knowlton, K. Eberhard, and J. Sedivy. 1995. Integration of visual and linguistic information during spoken language comprehension. *Science* 268:1632–1634.

Stefanie Tellex, Thomas Kollar, Steven Dickerson, Matthew R Walter, Ashis Gopal Banerjee, Seth Teller, and Nicholas Roy. 2011. Approaching the symbol grounding problem with probabilistic graphical models. *AI Magazine* 34(4):64–76.

Jesse Thomason, Shiqi Zhang, Raymond Mooney, and Peter Stone. 2015. Learning to interpret natural language commands through human-robot dialog. In *Proceedings of the 24th International Conference on Artificial Intelligence*. AAAI Press, IJCAI'15, pages 1923–1929. http://dl.acm.org/citation.cfm?id=2832415.2832516.

Peter D. Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *J. Artif. Int. Res.* 37(1):141–188. http://dl.acm.org/citation.cfm?id=1861751.1861756.