KIDE4Assistant: an Ontology-Driven Dialogue System Adaptation for Assistance in Maintenance Procedures

Cristina Aceta[1,*], Patricia Casla[1], Izaskun Fernandez[1] and Aitor Soroa[2]

[1] *TEKNIKER, Basque Research and Technology Alliance (BRTA), Iñaki Goenaga 5, 20600 Eibar, Spain*
[2] *HiTZ Center—IXA, University of the Basque Country (UPV/EHU), 20018 Donostia, Spain*

Abstract

Developing systems that facilitate natural interaction between humans and industrial systems by using current state-of-the art technologies, such as Deep Learning techniques, requires large amounts of training data, usually scarce in industrial scenarios, and the resulting systems are difficult to adapt to other settings. This paper illustrates an ontology-driven adaptation of KIDE4I, a generic semantic-based task-oriented dialogue system, to set up with a limited amount of effort KIDE4Assistant, a dialogue system for assistance in maintenance procedures through natural interactions. This adaptation approach bridges the gaps related to data availability and system reuse by exploiting expert knowledge together with existing ontologies and resources to obtain a fully-functional task-oriented dialogue system. Furthermore, the evaluation of KIDE4Assistant, performed through a user experimentation, shows the promising results of this approach.

Keywords

Ontology reuse, Task-oriented dialogue system, Industry 5.0, Human-machine interaction

1. Introduction

The introduction of innovative Industry 5.0 enabling technologies –such as AI or collaborative robotics– in industrial scenarios has reduced workers' physical workload. However, an increase of the cognitive load to control and manage such technologies has been observed [1]. The possibility of human workers communicating to industrial systems through natural language in the same way as to fellow colleagues is highly encouraged, as they do not have to memorize specific words or sequences to interact with them. This reduces workers' mental stress and leads to increase user acceptance [2]. However, to develop task-oriented dialogue systems that facilitate a natural interaction between humans and industrial systems by using current state-of-the art technologies –such as Deep Learning (DL) techniques– is a difficult task, and leads to a series of challenges that can be summarized in the following points:

- *Lack of training data.* The use of DL techniques to develop task-oriented dialogue systems requires great amounts of data for training. Moreover, currently available data is usually

---

FOMI 2022: 12th International Workshop on Formal Ontologies meet Industry, September 12-15, 2022, Tarbes, France

[*] Corresponding author.

cristina.aceta@tekniker.es (C. Aceta); patricia.casla@tekniker.es (P. Casla); izaskun.fernandez@tekniker.es (I. Fernandez); a.soroa@ehu.eus (A. Soroa)

0000-0003-3566-9460 (C. Aceta); 0000-0001-8496-5827 (P. Casla); 0000-0002-0762-1737 (I. Fernandez); 0000-0001-8573-2654 (A. Soroa)

CEUR Workshop Proceedings (CEUR-WS.org)

bound to specific domains and is also scarce [3], especially in industrial scenarios [4] and languages other than English.

- *Restricted natural language communication.* Since using modern technologies is conditioned by the lack of training data, most dialogue solutions designed for industrial settings are highly specific for the target application and only support rigid language [5, 6].
- *Limited adaptation capabilities.* Due to the high specificity of dialogue systems designed for industrial environments, their capacity to be reused in other scenarios is very limited and usually bound to expert manual work and high development time and costs [7].

KIDE4I [8] contributes to progress in all these challenges by taking advantage of the many benefits of Semantic Web Technologies by enabling an easy pathway to (i) define the domain of application in detail through ontologies and, thus, reducing ambiguity between the agents that take part in the interactions [9] and (ii) model the dialogue process through relationships between individuals and their properties. These characteristics lead to a generic natural interaction system that is easy to adapt to different use cases –as shown in the use case adaptations in [8]–, using a reduced amount of data and making human knowledge and logic available to computers.

This paper aims to show how to easily setup a dialogue system on an industrial scenario –in which workers receive assistance while executing maintenance procedures– by having, as a starting point, an initial implementation of KIDE4I that supports natural interaction for a logistics scenario and by reusing existing domain ontologies.

The rest of the paper is organised as follows: Section 2 presents the implementation of the methodology in [8] to adapt KIDE4I for the maintenance procedure assistance scenario, with special focus on the reuse of ontologies for domain ontology development. Section 3 describes the user study performed to evaluate the resulting system, KIDE4Assistant, and the results obtained. Finally, main conclusions are presented in Section 4.

2. KIDE4I Adaptation for Maintenance Procedure Assistance

KIDE4Assistant is the result of adapting KIDE4I to assist workers on maintenance execution procedures through natural interactions. This adaptation has been carried out according to the methodology presented in [8], which consists of the following main steps: (i) use case characterization, (ii) adaptation of the key element extraction component, (iii) ontology modelling and instantiation and (iv) adaptation of the dialogue manager. The work performed in each of these steps to develop KIDE4Assistant is reported in the following lines.

2.1. Use Case Characterization

This first step aims to identify the requirements to be considered in the different modules of the dialogue system, in terms of the elements included in the domain, the key elements that refer to them and the target system's functionalities that the system must deal with.

For the maintenance procedure assistance scenario, a set of maintenance procedures extracted from technical manuals have been considered to determine the requirements to guide users through the processes described in them. These procedures, which are in Spanish, are structured in *methods*, *tasks* and *steps*. *Methods* define different ways to perform the same procedure (e.g., in usual conditions or in a clean room). Each *method* has a set of *tasks* (e.g., extract a battery, install a battery), and each *task* consists of a set of *steps* (e.g., disconnect the machine, open the lid). Given a procedure, KIDE4Assistant would first request the user to select the method to follow (when it has more than one; if not, it is automatically selected). Then, the system would give the description of the current method, task and step. At this point, the user should be able to (i) navigate through the different elements of the procedure, (ii) ask for information repetition, (iii) restart a method or task and (iv) obtain other related information, such as the list of necessary tools to perform the procedure or a more extensive description in the form of text and images –which are shown in a screen so users can easily follow the information provided by the system. All natural interactions would be in Spanish, as it is the procedures' language.

2.2. Key Element Extraction Component Adaptation

The second step aims to adapt the component that is in charge of extracting the relevant key elements to interpret user commands. Considering the use case characterisation, the key elements to be identified by KIDE4Assistant are *actions* and *targets*, which correspond to the keywords used to determine the action to perform (see Example (1), for the *Show tool list* action).

(1) **Muéstrame**$_{action}$ la lista de **herramientas**$_{target-determineAction}$
Show$_{action}$ me the **tool**$_{target-determineAction}$ list.

In this case, as the key elements to identify are common with KIDE4I's generic version, its definitions, rules and scripts for key element extraction have been reused with no modifications.

2.3. Ontology Modelling and Instantiation

In this step, the necessary information to model the use case must be identified and instantiated into the TODO[1] ontology, the core element in KIDE4I. This ontology has been implemented in OWL, and it is available in multiple formats, such as RDF/XML, N-Triples, Turtle or JSON LD. For this use case, the format used was RDF/XML.

This step follows different phases, which are closely related to the different TODO modules, TODODom and TODODial [10]:

1. **Modelling and instantiation of the domain (TODODom)** to determine the *world elements* taking part in the interaction scenario (within the TODODW submodule) and the *actions* to be performed in the target system (within the TODODFA submodule), along with the expressions to refer to them and their associated machine-readable information.
2. **Modelling and instantiation of information related to dialogue management (TODODM submodule, part of the TODODial module)**, by defining the logic implications of the different outcomes of each interaction between the system and its users. Most of the elements in the KIDE4I initial implementation would be reused, and only a few new elements must be included, if necessary.

For the modelling of KIDE4Assistant's domain ontology, the system functionalities and 6 specific procedures –that follow the structure described in Section 2.1– have been considered

---

[1] https://w3id.org/todo

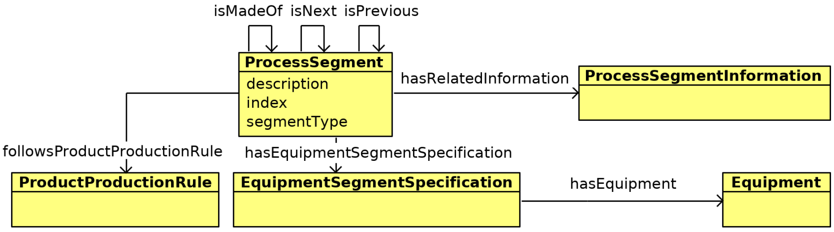

Figure 1: TODODW main classes and relations.

to define, according to the LOT (Linked Open Terms) methodology [11], the 19 competency questions (CQ) that TODODW should answer. Some examples of these CQs are the following:

- How many tasks does the method have?
- Which is the tool list?
- Which is the description of a procedure/method/task/step?

Taking these CQs as basis, an analysis of existing ontologies has been performed. The VAR [12] ontology has been selected to be reused to model in TODODW the world elements for the maintenance assistance scenario. VAR is intended to represent a workplace digital twin by including workplaces, processes and workers, along with the relations between them. The process descriptions in VAR fit the maintenance procedures described above and, thus, TODODW reuses this ontology as-is, with no further extensions/adaptations. The main classes and object properties are depicted in Figure 1, and their definitions can be found below:

- `ProcessSegment` correspond to the different parts a process can be divided in. In this case, this class includes procedures and their parts (i.e. *methods*, *tasks* and *steps*). To distinguish the type of segment (e.g., *method*), the `segmentType` data property is used.
- `ProductProductionRule` includes the elements that provide additional details on a `ProcessSegment`, such as images or documents.
- `ProcessSegmentInformation` models the description(s) of each `ProcessSegment`.
- `EquipmentSegmentSpecification` (ESS) groups the tools needed to perform a procedure following a specific method. These ESS are related to each tool (`Equipment`) through the property `hasEquipment`.

In order to validate the ontology and generate its ABOX, the 6 aforementioned procedures have been extracted from their corresponding manuals and formatted as JSON files, from which procedure information has been obtained. From this JSON, the relations between *procedures*, *methods*, *tasks* and *steps* have been instantiated, including sequential relations (e.g., *Step* 2 comes after *Step* 1 and before *Step* 3). Furthermore, for each method, their tool list – with its corresponding tool individuals– has also been instantiated and, for each structural element of the procedure (i.e., *methods*, *tasks* and *steps*), additional information and/or extended information has been automatically included. Finally, the elements that make reference to procedure parts –*procedures*, *methods*, *tasks*, *steps* and *tools*, which correspond to the key elements labelled as *targets*– have also been added. For the rest of the domain information, actions have been

Table 1
KIDE4Assistant: summary of total and, reused classes and automatic instances

	Dialogue		Domain		All	
	Total	**Reused/Auto**	**Total**	**Reused/Auto**	**Total**	**Reused/Auto**
Classes	82	80 (97.6%)	134	132 (98.5%)	216	212 (98.1%)
Instances	80	80 (100%)	584	516 (88.4%)	664	596 (89.8%)

manually instantiated, and related linguistic information has been obtained by following the population strategy described in [13]. More specifically, 22 frames, 50 frame heads and 187 lexical units have been semi-automatically instantiated.

For KIDE4Assistant's dialogue ontology modelling, only two classes have been added to TODODial to cover an additional interaction regarding method selection, which are not included in the generic version: *MethodOptionsResponse*, to inform the user that the selected procedure has more than one method and *MethodDisambiguationRequest*, to request the user to confirm that they want to follow the method in the request.

The instances for these classes have been manually instantiated, whereas the rest of individuals have been reused from KIDE4I's initial implementation. Indeed, Table 1 shows that nearly all the classes for this adaptation have been reused (a 98.1%), and an 89.8% of the instances have been obtained through automatic methods or reused from the initial version.

2.4. Dialogue Manager Adaptation

All of the dialogue information has been reused; however, minimal modifications on the dialogue manager's logic have been needed to cover the specific necessities of the use case that were not included in the generic initial version (i.e., additional interactions to disambiguate the method to follow to carry out the procedure).

3. KIDE4Assistant Evaluation

To evaluate KIDE4Assistant's performance, a user study has been carried out. Given a set of specific maintenance procedures that were given to them, users had to interact with KIDE4Assistant to get assistance during their execution by navigating along the different steps and, whenever necessary, asking for additional information. An example of a possible interaction between users and KIDE4Assistant is reported below:

- USER: "The robot's battery is damaged"
- SYSTEM: "The method to perform this procedure has 2 tasks. The first task is 'Removing the battery', and consists of 6 steps. Use this procedure to remove the battery."
- S: "The 1^{st} step is 'Disconnect the power, hydraulic pressure and pressurised air connections!' "
- U: "Okay, now what?"
- S: "The next step, the 2^{nd} of 6, is 'Always cut off the paint with a knife and sand the painted edge when disassembling the robot parts.' "

A total of 12 participants, with ages ranging from 24 to 59 and different levels of expertise in the field, were involved. Each of them interacted with KIDE4Assistant to complete two different procedures. The evaluation aimed to assess (i) whether the interaction goal was fulfilled (dialogue completion), (ii) how many turns did it take for that interaction goal to be reached (dialogue steps), and (iii) the system's response time for each interaction.

To assess dialogue completion, the 294 dialogues generated in the evaluation task have been analysed by a group of experts, who were expected to determine whether the user goal was successfully *completed* or *not completed*. In the case of *completed* tasks, dialogues were classified between *fully completed* or *partially completed*, depending on whether the user had to reformulate the query at some point of the dialogue. The results are very positive: the successful completion rate is 78%, where only a 7% of these dialogues were classified as *partially completed*.

As for the rest of evaluated characteristics, the average number of steps required to complete each dialogue was 1.5, and the system took an average of 1.88 seconds to respond to a user request. These numbers are positive enough to determine that KIDE4Assistant allows an agile interaction between users and the system.

4. Conclusions

This paper shows the adaptation process of KIDE4I, a generic semantic based task-oriented dialogue system, to set up KIDE4Assistant, a maintenance procedure assistant, specially focusing on the reuse of existing ontologies and resources (such as datasets or libraries) for delivering natural interaction for maintenance assistance.

To perform the adaptation and get the KIDE4Assistant task-oriented dialogue system, the methodology defined in [8] has been used, first characterising the scenario and then adapting each module in KIDE4I, making this process as simple as possible by reusing different resources:

- For key element extraction: reuse of most rules from the generic implementation.
- For ontology modelling: reuse of TODO[10] and VAR[12] for TODODW modelling.
- For ontology instantiation: for TODODM, reuse of instances from the generic implementation; for TODODom, reuse of linguistic resources by using the population strategy in [13] and automatic instantiation of procedures from technical manuals in JSON format.
- For the dialogue manager design: reuse of most of the functions from the generic implementation, with slight modifications required.

In this sense, it has been shown that, once a generic implementation is obtained, the effort to adapt KIDE4I to other use cases is considerably reduced, mainly motivated by the use and reuse of existing ontologies and resources. Nevertheless, it is still necessary to manually model and instantiate some classes and individuals.

The evaluation of KIDE4Assistant reported in Section 3 shows that the success rate of completed dialogues, the number of necessary steps to complete a dialogue, and the system's response time are promising. With this, the ontology-driven KIDE4I and the adaptation strategy could be considered as a good starting point to easily setting up a dialogue system that enables natural interaction in industrial environments, reducing considerably the main gaps of other state-of-art techniques such as lack of data, or highly specific solutions thanks to the use of the semantic technologies.

Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement 958371, project OntoCommons.

References

[1] M. Madonna, L. Monica, S. Anastasi, M. Di Nardo, Evolution of Cognitive Demand in the Human-Machine Interaction Integrated with Industry 4.0 Technologies, WIT Transactions on The Built Environment 189 (2019) 13–19.

[2] J. Kildal, I. Fernández, I. Lluvia, I. Lázaro, C. Aceta, N. Vidal, L. Susperregi, Evaluating the UX Obtained from a Service Robot that Provides Ancillary Way-Finding Support in an Industrial Environment, in: Advances in Manufacturing Technology XXXIII: Proceedings of the 17th International Conference on Manufacturing Research, 10-12 Sep 2019, Belfast, volume 9, IOS Press, 2019, p. 61.

[3] P. Budzianowski, T.-H. Wen, B.-H. Tseng, I. Casanueva, S. Ultes, O. Ramadan, M. Gašić, MultiWOZ–A Large-Scale Multi-Domain Wizard-of-Oz Dataset for Task-Oriented Dialogue Modelling, arXiv preprint arXiv:1810.00278 (2018).

[4] A. Luckow, M. Cook, N. Ashcraft, E. Weill, E. Djerekarov, B. Vorster, Deep Learning in the Automotive Industry: Applications and Tools, in: 2016 IEEE International Conference on Big Data (Big Data), 2016, pp. 3759–3768. doi:`10.1109/BigData.2016.7841045`.

[5] G. Bugmann, J. N. Pires, Robot-by-voice: experiments on commanding an industrial robot using the human voice, Industrial Robot: An International Journal (2005).

[6] G. Veiga, J. Pires, K. Nilsson, Experiments with service-oriented architectures for industrial robotic cells programming, Robotics and Computer-Integrated Manufacturing 25 (2009) 746–755.

[7] D. Jurafsky, J. H. Martin, Speech and Language Processing (Draft), Hentet, 2020.

[8] C. Aceta, I. Fernández, A. Soroa, KIDE4I: A Generic Semantics-Based Task-Oriented Dialogue System for Human-Machine Interaction in Industry 5.0, Applied Sciences 12 (2022). URL: https://www.mdpi.com/2076-3417/12/3/1192. doi:`10.3390/app12031192`.

[9] D. Antonelli, G. Bruno, Human-Robot Collaboration Using Industrial Robots, in: 2nd International Conference on Electrical, Automation and Mechanical Engineering, Atlantis Press, 2017, pp. 99–102.

[10] C. Aceta, I. Fernández, A. Soroa, TODO: A Core Ontology for Task-Oriented Dialogue Systems in Industry 4.0, in: Further with Knowledge Graphs, IOS Press, 2021, pp. 1–15.

[11] M. Poveda-Villalón, A. Fernández-Izquierdo, R. García-Castro, Linked Open Terms (LOT) Methodology, 2019. URL: https://doi.org/10.5281/zenodo.2539305. doi:`10.5281/zenodo.2539305`.

[12] I. Fernández, P. Casla, I. Esnaola, L. Parigot, A. Marguglio, Towards Adaptive, Interactive, Assistive and Collaborative Assembly Workplaces through Semantic Technologies, in: Proceedings of the 10th International Conference on Interoperability for Enterprise Systems and Applications (I-ESA 2020), 2020.

[13] C. Aceta, I. Fernández, A. Soroa, Ontology Population Reusing Resources for Dialogue

Intent Detection: Generic and Multilingual Approach, in: Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021), 2021, pp. 10–18.