



TALKING MACHINES

VOICE ASSISTANT FOR MACHINE INTERACTION IN INDUSTRY 4.0

MOTIVATION / STATE OF THE ART

Today we are in the fourth industrial revolution or also known as Industry 4.0, where virtual and physical manufacturing systems cooperate with each other in a flexible way on a global level (Pistorius, 2020, p.5). Meanwhile the recent advancements in AI offer significant chances for the improvement and extended use of virtual assistants. As it is the base for natural language processing (NLP), a key technology part of chatbots utilized to understand the human language (Market Research Future, 2021). There are many virtual assistants on the market for home automation and personal tasks (Petrock, 2020). But the global voice assistant market is still expected to grow significantly in the next 5 years. One of the key drivers in Europe being the adaption of voice-based-solutions in the industrial sector (Market Research Future, 2021).

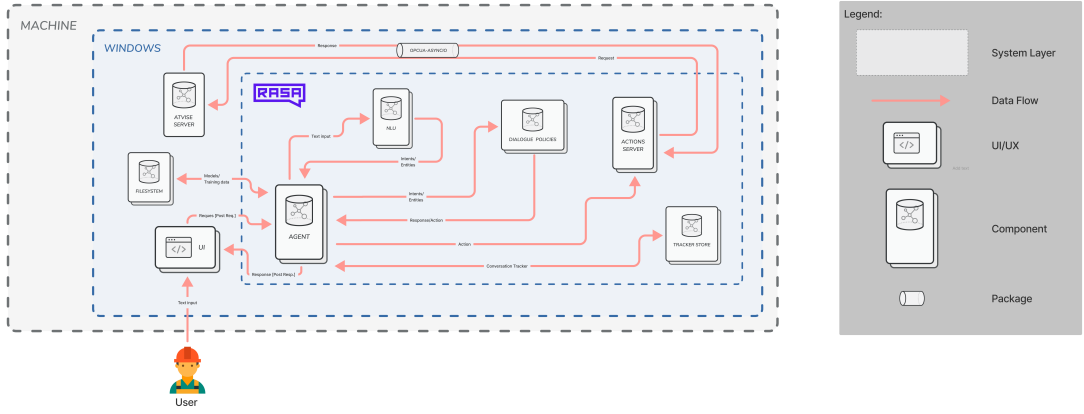
GOAL

Our project aimed to explore the chances of voice assistants for Industry 4.0 in HEKUMA machines, as this project is industry cooperation between the HEKUMA GmbH and the HSA. The HEKUMA GmbH is a leading special machine construction company from the Munich area. The overall goal was archived by first researching the state of the art, evaluating different approaches, and then choosing a fitting framework for a future prototype. Next the architecture of that said prototype was designed and later implemented. During this process we oriented ourselves on, example conversations provided by the HEKUMA GmbH. Adapted versions of those can be seen in the “Results” section below.

ARCHITECTURE

The design of our concept is based around the Rasa framework. Rasa was chosen amongst other reasons for the fact that it can run locally on the machine without needing to have an internet connection. With a very limited amount of training data in form the form of classified example sentences, Rasa makes it very easy to analyze intents, extract entities and connect them to answers or possible actions. These actions are defined through Rasa Actions and make it possible for us to attach our chatbot to an industrial machine. As we were only supposed to develop a prototype, the industrial machine in our case is only represented by an atvise server. It contains information about the state of an imaginary machine, with alarm values and other important operating information. Changing values on this atvise server does, at this stage, not lead to any changes on a real machine. But it was incredibly useful for prototyping. The atvise server exchanges data with our Rasa actions server through OPC UA. OPC UA is a machine to machine communication protocol. Therefore, our Rasa actions server needs to understand this form of communication. Which was made possible by using the library opcua-asyncio. opcua-asyncio is a Python 3 library and worked perfectly well with our actions Rasa actions server, which also needs to be written in Python.

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.



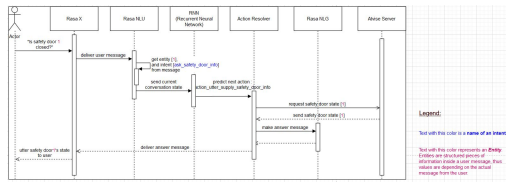
Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

TEAM

Saliu Bah Marco Lenz
Cristina Vizan Olmedo Jung Eun Park

DETAILS

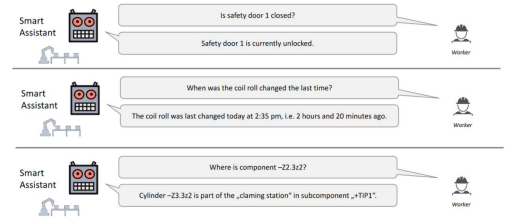
To go further into detail, concerning the exact process of a chatbot interaction, one can have a look at the following sequence diagram.



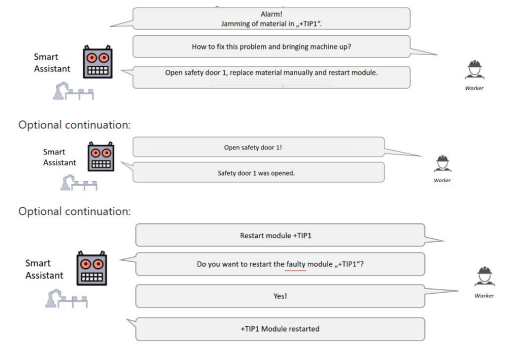
In this example a worker wants to know the state of a safety door on the machine. They input their command through Rasa X which is a chat UI already integrated into Rasa. The different components integrated into Rasa then take over and deciphers the user's intent and extracts the exact door number. Next Rasa actions contacts the atvise server to find out about the specified safety door state. Finally the answer is passed back to the worker.

RESULTS

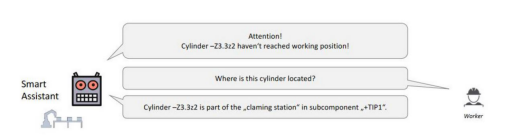
The talking machines team was able to implement the following use cases in English. Simple state requests:



Proactive communication, optional continuations and processing contextual information:



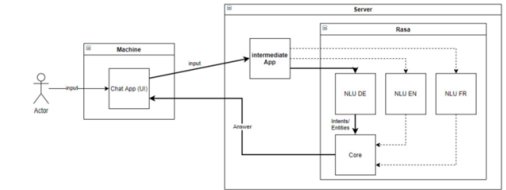
Proactive communication and processing contextual information:



OUTLOOK

Next steps for the project could be implementing this prototype on a real Hekuma machine. This would mean instead of just flipping values on the atvise server, interaction with the chatbot could move actual machine parts. Implementation of this would need an adaption of the current implementation to the specific machine. For example, you would need to verify if information about the state of the machine is structured in a similar way as it is on our atvise server model. Still, the overall architecture of the prototype would make this possible, as it is tightly fitted to the requirements of a Hekuma machine. Further, the prototype is effortlessly expandable. A variety of a new use cases can easily be implemented.

A completely new feature to implement, could be multilingualism. We already developed a draft for this during our time on the project, having a separate NLU for each desired language, as shown in the image.



REFERENCES

1. Market Research Future. (2021, October). *Voice Assistant Market Size, Share Forecast 2027*. <https://www.marketresearchfuture.com/reports/voice-assistant-market-4003>
2. Petrock. (2020) *Voice Assistant and Smart Speaker Users 2020*. eMarketer. <https://www.emarketer.com/content/voice-assistant-and-smart-speaker-users-2020page-report>
3. Pistorius, J. (2020) *Industrie 4.0 – vierte industrielle Revolution*. Springer Vieweg