COMP[39]900 – 2019 T1

Project 1

X-o-Bot: A Generic Dialog Framework for Conversational Agents

Conversational agents or chatbots have become very common for websites and corporate services to help customers/users quickly find answers to their questions without depending on a customer support person and possible waiting queues. Advances in Machine Learning, Cognitive Services and IoT (Internet of Things) made it possible to diffuse the technologies in the home users' segment at very low cost.

Similar principles can be applied to education, with courses and students' support as possible outlets for the technology: students often have simple questions that do not need to rely on the instructor to provide answers for. The answers are often already available in the corpus of data available for that course or in the knowledge database for a specific service.

One striking feature of the existing services is that they are extremely dumb and are only suitable for very simple question/answer interactions. More recent technology advances showed that the conversations can be somewhat designed enabling more realistic or believable interactions (see Google Dialogflow as an example).

This project focuses on the creation of a generic and modular architecture for conversational agents which is based on three core components:

1. A knowledge extraction/building module (which includes search, ingestion and knowledge representation)
2. A dialog builder (which can leverage on existing APIs and theories about interaction)
3. An Application/UI component which allows to 'plug' the agent into multiple channels (from the simple Web chat, using other communication tools and other environments/tools – from games to VR, from assistants such as Google Assistant or Amazon Alexa to robotic interfaces)

You will have to choose an application domain among the following:

- A household company assistant (targeting the elderly group by, for instance, asking them general wellbeing questions – hence an appropriate set of user stories will be required)
- A student-tutor support system (extracting all content from a specific course and answering key questions)
- A household butler, able to coordinate devices and services and provide the ability to have a meaningful interaction with the user (not just the Alexa skills!)

A Python-based solution is preferred (but not required) using:

1) modern development frameworks,

2) cloud-based architectures, and

3) building a solution which leverages on existing services/APIs without creating a full dependency on the selected tools.

It is essential that system requirements are refined with the stakeholders and that the engineering of the system is thought through properly before building the individual components.

Additional material for Topic 1

Conversational Agents

1. Architecture for building Conversational Agents that support Collaborative Learning-CMU.pdf
http://www.cs.cmu.edu/~./emayfiel/application_papers/KumarAndRoseInPress.pdf

2. Dialogue and Conversational Agents,
http://courses.washington.edu/ling575/SPR2015/slides/ling575_class1_SDS_intro_short.pdf

3. Conversational Agents, Dan Jurafsky Stanford University
https://web.stanford.edu/class/cs124/lec/chatbot.pdf

4. Dialogflow - Build natural and rich conversational experiences
https://dialogflow.com