netidee
STIPENDIEN

Dialogical Access to Lightweight Semantic Web Services

Endbericht | Call 13 | Stipendium ID 3451

https://www.netidee.at/dialogical-access-lightweight-semantic-web-services

Inhalt

1 Einleitung

This final report provides insights regarding the latest status of the dissertation Dialogical Access to Lightweight Semantic Web Services. We first introduce the topic, motivation and the research questions in Section 2. In Section 3, we explain the existing results and later in Section 4 we describe what is left to accomplish in order to answer all research questions. Section 5 provides some future directions for everyone who would like to utilize our results.

2 Allgemeines

Knappe Zusammenfassung der Arbeit, Fragestellung, Ziel der Arbeit

Thanks to the development in the Artificial Intellligence, especially in Machine Learning in recent years, conversational agents (e.g., Amazon Alexa, Google Assisstant) are becoming the main way of consuming services on the web. These systems typically connect to a knowledge source or a hard coded set of web services to enable users to achieve certain tasks to reach their goal. These dialog systems are powered by Natural Language Understanding units, that classify user utterances to certain intents. Although these intents typically reflect the functionality of the web services the dialog system utilizes, they are tightly coupled, therefore it is not straightforward to extend these dialog systems with new web services.

In this thesis, we address the flexible expansion of goal-oriented dialog sytems (i.e. the dialog systems whose main purpose is guiding users to complete a goal) with new services with the help of lightweight semantic annotations of web services. We try to answer the following questions:

To what extent schema.org actions can be used as a lightweight semantic web services vocabulary:

Schema.org is a de facto industrial standard vocabulary annotating resources on the web. It also contains an actions subset, which we argue that may be suitable for annotating web services. The actions subset of schema.org provide a generic set of actions, but how much can it support lightweight semantic web services annotation? What are the necessary aspects to describe, and how many of those can schema.org actions support? How can the schema.org and its actions subset be customized for the web service annotation task?

To what extent we can generate goal-oriented dialog systems based on semantic web service descriptions:

By its design, schema.org contains its semantics partially in the names and descriptions of types and properties. How much of a dialogue can be generated by processing the semantics that is implicit in the action annotations? Is there a mapping between schema.org actions and intents in a goal-oriented dialog system? Can linked resources provide a basis for generating a dialog on the fly?

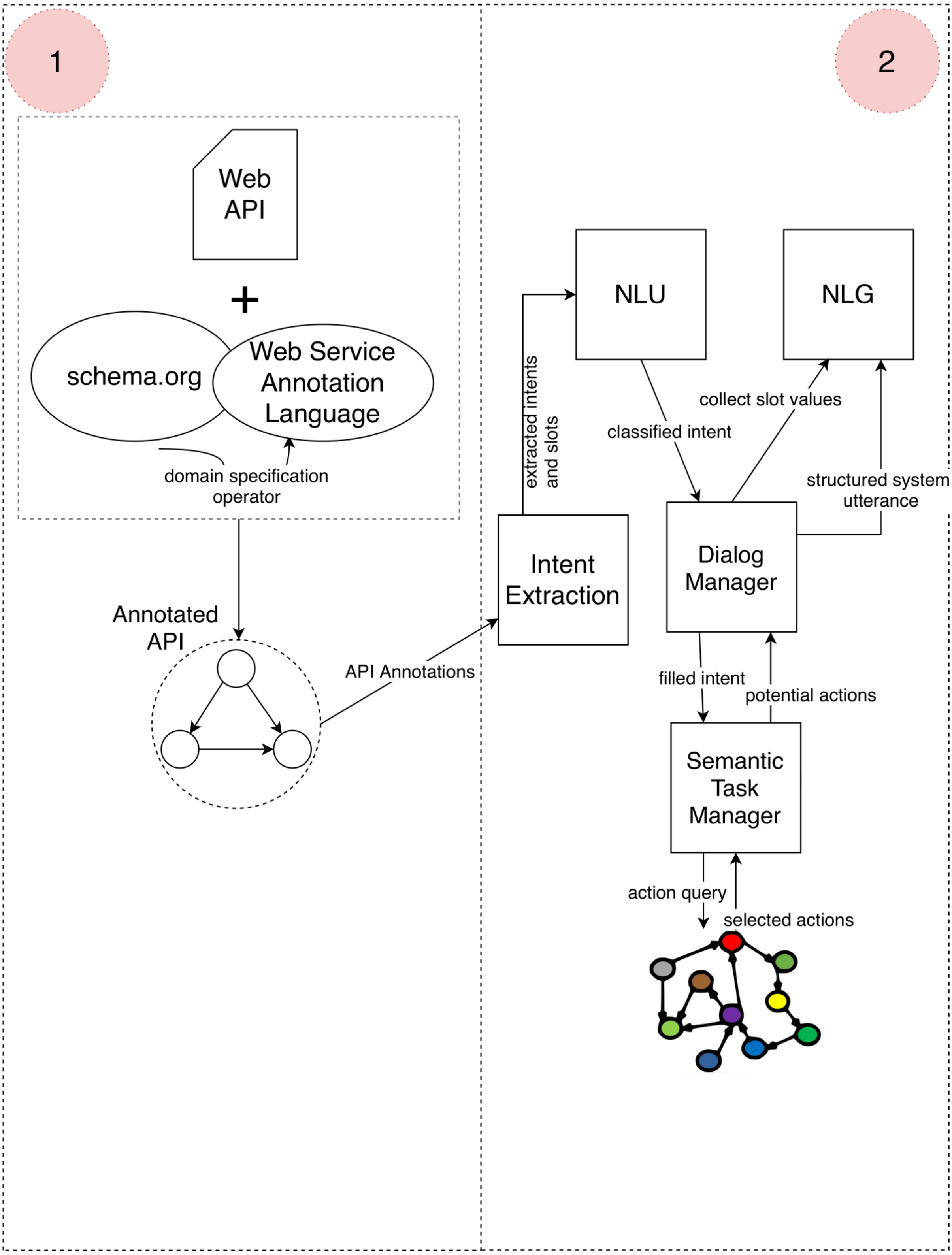

Figure 1: An overview of DIALSWS approach

Figure 1 summarizes the overall process of using annotated web services for semi-automatically generating goal-oriented dialogs. The upper half of the figure (1) depicts the web service annotation process and the lower half (2) how they are consumed. To answer our research questions, we first analyze schema.org actions in the scope of lightweight semantic web services literature and extend and restrict the vocabulary when necessary. For this, we use the domain specification process in order to make schema.org suitable for the web service annotation task in various domains. The domain specification process restricts and extends a vocabulary in order to make it suitable for specific domains and tasks. We support this process by providing guidelines and tools for annotation of web services. An annotated web service (i.e., a set of actions) is then used to extract intents and their slots that need to be filled by a user. The intent extractor is connected to the NLU unit of a goal-oriented dialog system. After a user utterance is classified to an intent, the dialog manager collects the inputs for necessary slots passes the filled intent to the semantic task manager. The task manager queries the web service annotations to retrieve the relevant actions. After an action is completed, the task manager extracts any potential actions and allows the dialog to continue and complete the task.

Our core contribution is to increase the flexibility and openness of goal-oriented dialog systems. This is possible based on the assumption that the core of a goal-oriented dialog systems reflect the capabilities of the backend systems to which they are connected. We exploit this assumption by using semantically annotated web services. The main challenge to achieve our goals is that there is a significant effort required for semantically describing web services. This effort causes the so called chicken-egg problem. There are no service annotations because there are no applications and there are no applications because there are no annotations. We aim to overcome this hurdle, by creating a lightweight approach based on schema.org, which is a very widespread vocabulary. Our approach particulary utilizes the actions subset of schema.org and sees web services as a collection of operations that can be taken on resources.

3 Ergebnisse

Beschreibung der erreichten Ergebnisse

- An analysis of schema.org actions vocabulary in the context of semantic web services
- A concept and formalization for the domain specification process: We developed a domain specification process and formalized it while reusing SHACL components,a W3C standard for defining data shapes.
- The WASA Language and tools support for the lightweight of annotation web services: We provided a language by applying the domain specification process on schema.org for annotating web services. The documentation can be found at http://wasa.cc
- The Unified Intent Model and its implementation Unified Intent Model Ontology (UIMO) for generic description of dialog intents across different dialog system frameworks. The specification is published on http://dialsws.xyz

4 Geplante weiterführende Aktivitäten

Kurze Angabe von noch offenen Aktivitäten (Zeitplan)

- July-August 2020:
  - Implementation of intent generator
  - Implementation and evaluation of semantic task manager
- September 2020: Proof-reading and completion of the thesis

5 Anregungen für Weiterführung durch Dritte

Welche Weiterführungen für Dritte ergeben sich durch Ihre Arbeit?

We encourage third parties to create action annotations to expose their services on the web. The action annotations are directly compatible with schema.org annotations of "things" on the web, and can be easily stored together with the data in a Knowledge Graph. We provide tooling support for creating action annotations from scratch, and annotating existing HTTP APIs with lifting and grounding support.

Our work provides two outcomes that are decoupled, but at the same time complementary. The action annotations can be used with any kind of intelligent client as a way of decoupling client and web services. On the other hand, the actions can be processed and represented as intents for goal-oriented dialog systems. The generic intent model we propose can be used to implement more flexible goal-oriented dialog systems that are adaptable to different component implementations (e.g. intents can be easily migrated between different NLU vendors). We urge interested parties to represent their intents in our generic format and publish on the web to create a distributed intent knowledge graph, so people with similar needs can use them to kickstart their own goal-oriented dialog systems.