Data Engineering Course

Project Title

Semantic Web Service Discovery

Prepared By:

Ezdehar Jawabreh

Supervised By:

Prof. Mustafa Jarrar

Introduction

- Web Service is : is a software system designed to support interoperable machine-to-machine interaction over a network [1]

- They have many characteristics such as: being self-contained, reusable, composable, and highly interoperable.

- A user select a Web service with respect to their functional and nonfunctional requirements (QoS).

- The number of published Web services is increasing every day.

- This increasing number hinder many service related tasks such as: Discovery, Selection and Composition.

Project Idea

- A well known solution that address this challenge is: employing the semantic Web in Service related tasks, resulting in so-called **Semantic Web services.**

- Studies reveal that the use of semantic service selection reduced the search space and achieved better discovery results than conventional approaches.

- After reviewing and understanding state-of-art studies in this field we suggested and implemented the following idea:

Implementing an approach to automate discovery of Semantic Web services.

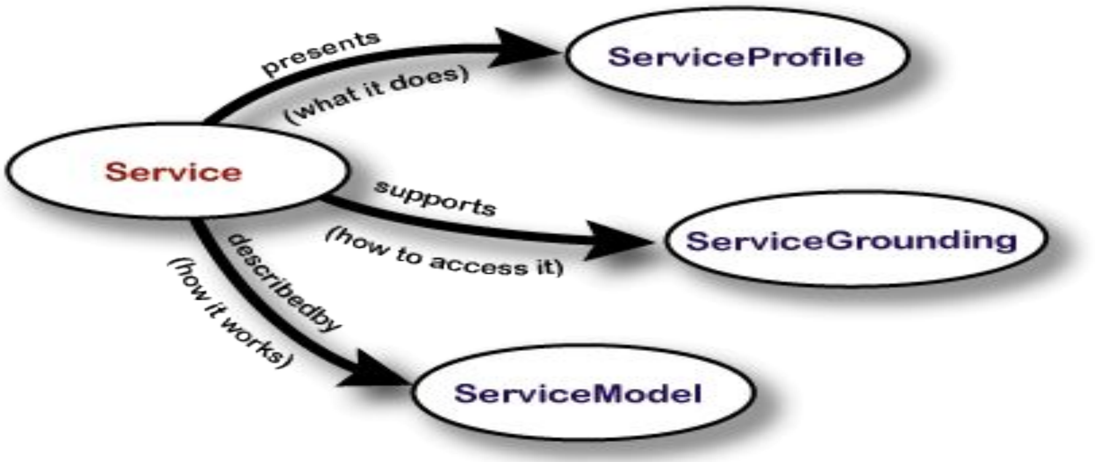
Background: Basics of Semantic Web Services

- Web Services are currently described using protocols such as: WSDL , UDDI and SOAP.
- These protocols are XML files that describe services and their functionalities syntactically without semantics .
- Based on them service selection is done using keyword matching in the service registries. which may result in incorrect selection of service [1].
- However semantic Web services is depending on ontologies to annotate Web services in a way that makes them more interpretable and self-describing, such as OWL-S, WSMO, WSDL-S, and SAWDSL.

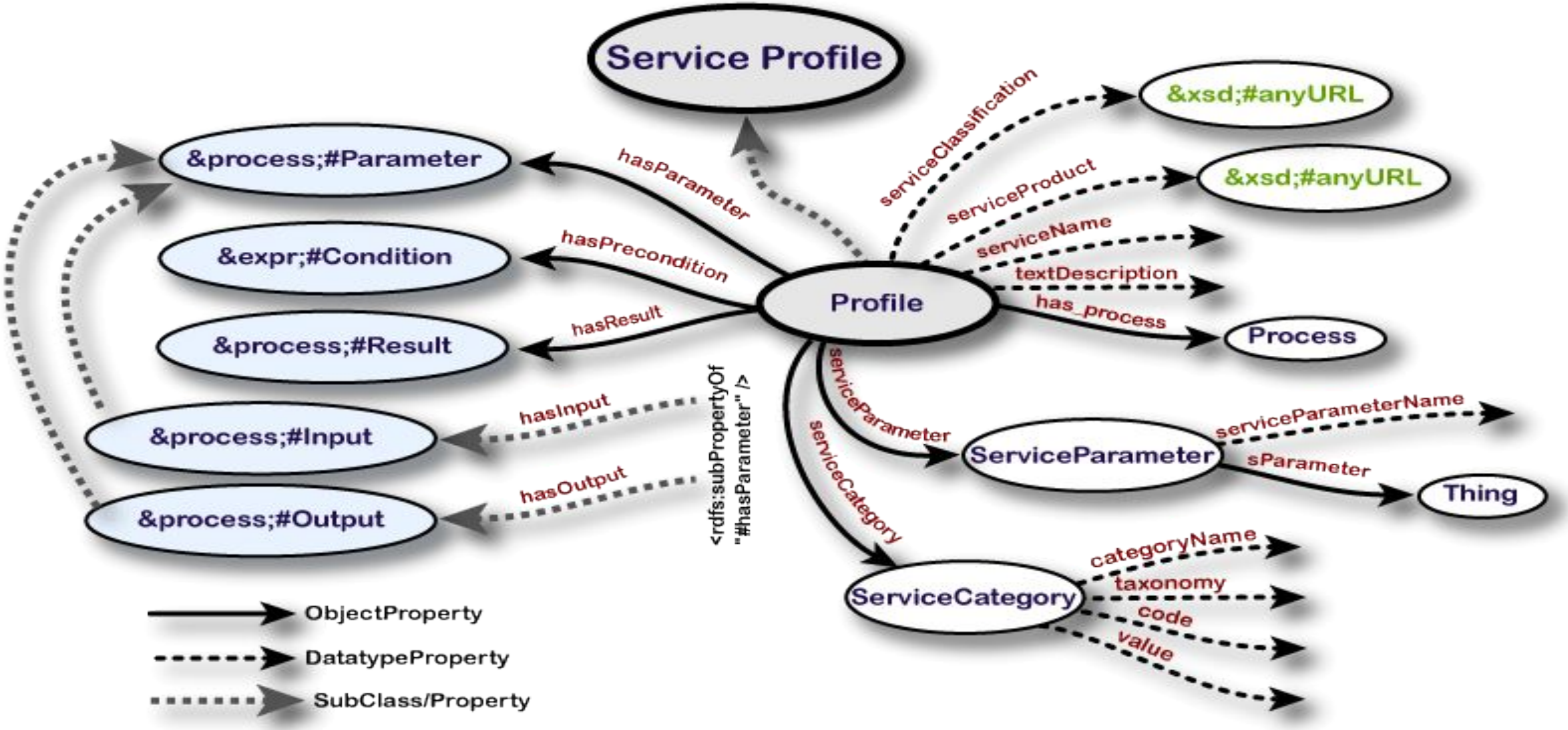
Background: OWL-S

It provides ontology to describe the services as following:

- **Service profile**: for advertising and discovering services;
- **Process model**: which gives a detailed description of a service's operation.
- **Grounding**: which provides details on how to interoperate with a service



Background: OWL-S

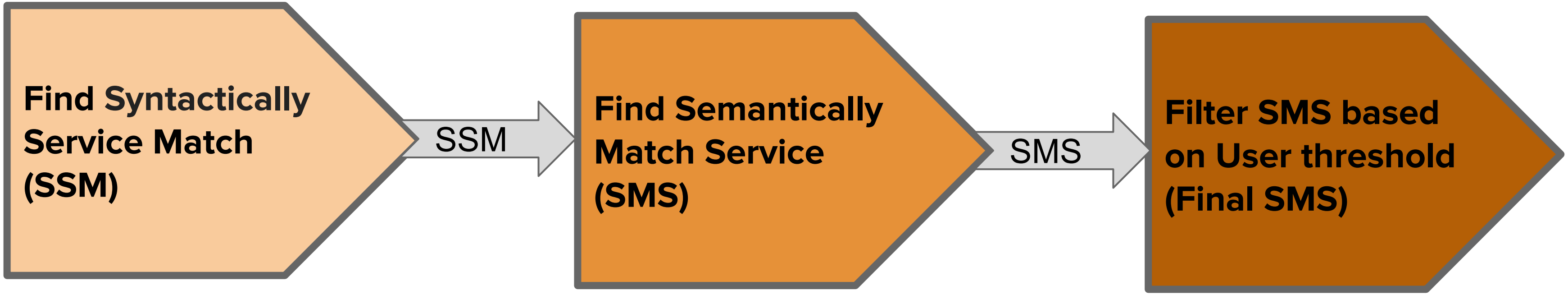


Proposed Algorithm

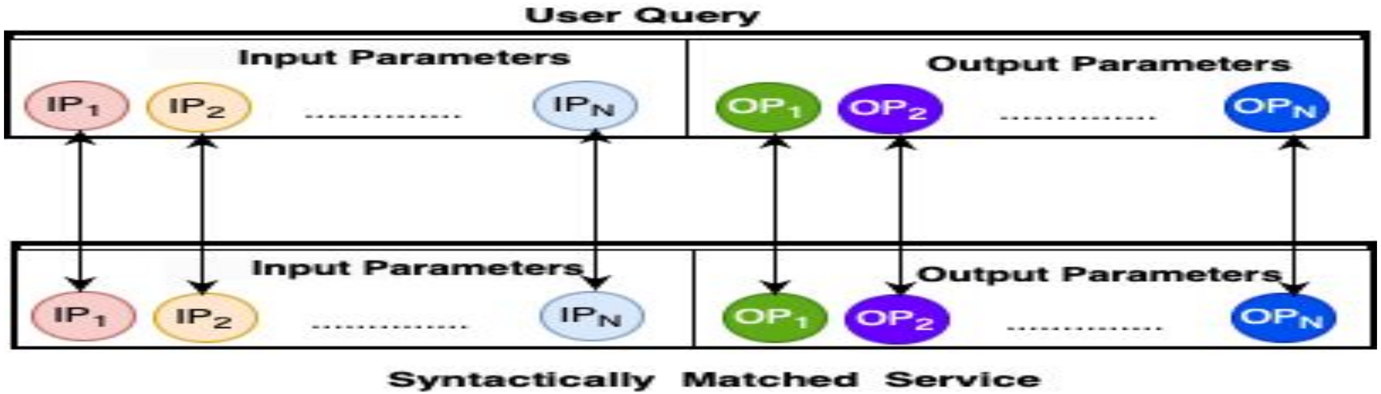
- Semantic service discovery algorithm, adapted from a recent study [2] with modification.
- It discover semantic services that match with a given user query.
- user query describe services syntactically by providing information about input and output parameters.
- Our algorithm follow both syntactically and semantically procedures to find semantic matched services.
- It utilize two types of ontologies :
  - OWL-S of Web services in Service Registry.
  - WordNET Ontology

Proposed Algorithm.Cont..

It consists of three steps



Semantic Filtration Step



$$FInputSim = \frac{1}{count(IP)} \sum_{IP \in a,q} Similarity\_Degree(IP_a, IP_q) \qquad (1)$$

$$FOutputSim = \frac{1}{count(OP)} \sum_{OP \in a,q} Similarity\_Degree(OP_a, OP_q) \qquad (2)$$

$$FinalSim = \frac{1}{2}(FInputSim + FOutputSim) \qquad (3)$$

Semantic Match Rules

Table 1: Semantic Match Rules

Semantic Match Rule	Similarity Degree
Exact Match	1.0
Non-Exact Match	WordNet Output
Direct PlugIn	0.9
Direct Subsume	0.8
No Match	0

Proposed Algorithm : Third Filtration step

A user-predetermined threshold filters the syntactically matched service, according to their final similarity, and a set of Semantically Matched Services (SMS) is returned to the user.

$$SMS = \{S | FinalSim(S) \geq SimThersold\}$$

Experiments and Results

Test Suite Preparation

We prepared a test suite from Web services published on SWS Test Collections
The test suite contains 50 Web services from different domain.

Domain	No of Web Services
Communication	7
Weapon	8
Education	8
Travel	8
Medical	6
Economy	5
Geography	3
Food	5
Total Number	50 Services

Service Regitsey Creation

```
PREFIX profile: <http://www.daml.org/services/owl-s/1.1/Profile.owl#>
PREFIX process: <http://www.daml.org/services/owl-s/1.1/Process.owl#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX service: <http://www.daml.org/services/owl-s/1.1/Service.owl#>


select  ?Service_Name
where {
            ?x profile:serviceName ?Service_Name.

     }

```

Table | Raw Response | Pivot Table | Google Chart

Chart Config

Service_Name
BookPrice
AcademicBookNumberOrISBNSearch
AcademicBookNumberSearch
FoodPriceService
GroceryStoreFoodService
RawFoodPrice Service
InvestigatingFinding
HistoricalPredicting
MedicalClinic Experiment service
Ben service
MAK service
BookFinder
VillageHotelService
HotelInfoService
Time-Measure Country City Hotel InfoService
SportsBeachService

```
select  ?Input_Parameter
where {
            ?x profile:hasInput ?Input_Parameter.

     }
```

```
select  ?Output_Parameter
where {
            ?x profile:hasOutput ?Output_Parameter.

     }
```

Results: Experiment 1

Experiment: 1 Exact Match	
User query: Inputs: sport, Output:destination ,Threshold:0.8	
Results	
Number of SMS	3
Service Name	Similarity Degree
1.sports_destination_service.owls	1.0
2.activity_city_service.owls	0.82
3.activity_town_service.owls	0.82

Results: Experiment 2

Experiment 2: Non-Exact User Match	
User query: Inputs: **automobile**, Output: **price**,Threshold:0.8	
Results	
Number of SMS	2
Service Name	Similarity Degree
1.car_price_service.owls	1.0
2.vehicle_price_service.owls	0.91

Results: Experiment 3

Experiment 3: Direct PlugIn Match	
User query: Inputs: **title**, Output: **comedyfilm**,Threshold:0.8	
Results	
Number of SMS	3
Service Name	Similarity Degree
1.title_comedyfilm_service.owls	1.00
2.title_film_service.owls	0.95
3.title_filmP2P_service.owls	0.95

Results: Experiment 4

Experiment 4: Direct Subsume Match	
User query: Inputs: **title**, Output: **film**,Threshold:0.8	
Results	
Number of SMS	4
Service Name	Similarity Degree
1.title_film_service.owls	1.00
2.title_filmP2P_service.owls	1.00
3.title_actionfilm_service.owls	0.85
4.title_comedyfilm_service.owls	0.85

Results: Experiment 5

Experiment 5:General Match Experiment	
User query: Inputs: **retailstore**, Output: **food, quality**,Threshold:0.8	
Results	
Number of SMS	2
Service Name	Similarity Degree
1.retailstore_foodquality_service.owls	1.00
2.retailstore_foodquantity_service.owls	0.92

State-of-Art

Study	Approach	Year
A Novel Approach for Semantic Microservices Description and Discovery Toward Smarter Applications [2]	provide an approach to Semantically describe microservices, used domain ontologies to compute input and output similarities according to user request	2023
A service composition evolution method that combines deep clustering [3]	use deep clustering (encoder-decoder) to cluster services according to functional and nonfunctional requirements using OWLS file.	2023

State-of-Art

Study	Approach	year
Building a semantic web services ontology in the pharmaceutical field using the OWL-S Language [4]	the construction of pharmaceutical ontological resources and adding semantic to the pharmaceutical web services.	2021
Recent Advancements in Semantic Web Service Selection.[5]	Discusses the semantic similarity score of the candidate web services according to user query. Discuss ontological solution for similarity calculations	2022
Req-WSComposer: a novel platform for requirements-driven composition of semantic web services.[6]	It uses semantic similarities between OWL-S files of Web services and user query on both user functional and nonfunctional requirements.	2021

Conclusions

- We proposed an algorithm that automated the discovery of semantic Web services.
- Our algorithm considered both the syntactical and semantically match with the user query.
- It computed the semantic similarity with user queries at the level of input and output parameters.
- it utilized two types of ontologies including the OWL-S ontology of the Web services and the WordNet ontology.
- We validated the algorithm by performing experiments on a service registry created from publicly published services and we gained satisfactory results with full coverage.

References

[1] https://www.w3.org/TR/ws-arch/

[2] Surianarayanan, Chellammal, Gopinath Ganapathy, and Pethuru Raj Chelliah. "A Novel Approach for Semantic Microservices Description and Discovery Toward Smarter Applications." In *Machine Intelligence and Smart Systems: Proceedings of MISS 2021*, pp. 89-101. Singapore: Springer Nature Singapore, 2022.

[3] Lu, Jiawei, Jiahong Zheng, Zhenbo Chen, Qibing Wang, Duanni Li, and Gang Xiao. "A service composition evolution method that combines deep clustering and a service requirement context model." *Expert Systems with Applications* 224 (2023): 119920.

[4] Fayçal, Zaidi, and Tari Abdelkamel. "Building a semantic web services ontology in the pharmaceutical field using the OWL-S Language." *2021 International Conference on Information Systems and Advanced Technologies (ICISAT)*. IEEE, 2021.

[5] Pahariya, Riddhi, and Lalit Purohit. "Recent Advancements in Semantic Web Service Selection." *IETE Journal of Research*(2022): 1-10.

[6] Driss, Maha, et al. "Req-WSComposer: a novel platform for requirements-driven composition of semantic web services." *Journal of Ambient Intelligence and Humanized Computing*(2022): 1-17.