International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys.14, No.1 (Oct-23)

http://dx.doi.org/10.12785/ijcds/140194

Agent-Based Model for Interpreting a Client Query as Part of Composite Web Service Discovery

Karima Belmabrouk[1] and Latifa Dekhici[2]

[1,2]*Computer Science,University of Science and Technology of Oran USTO-MB, Oran, Algeria*

Received 18 Feb. 2023, Revised 30 May. 2023, Accepted 30 Jul. 2023, Published 1 Oct. 2023

Abstract: The use of new tools, such as ontologies and agents, is becoming increasingly important in Web services research as they can help us analyze requests and conduct semantic searches for Web services. The main goal of this study is to propose a solution for intelligently interpreting user queries when searching for Web services. The proposed model involves the collaboration of two primary agents: a reactive agent and a cognitive agent. The reactive agent analyzes the user's query and the cognitive agent reasons based on a relevant domain ontology, matching the concepts extracted from the query with the services already published in the UDDI registry. This approach allows for the intentions and desires of the user to be understood, and returns the most suitable service accordingly.

Keywords: Web Services Discovery, Query analysis, Matching, Ontology, Agent

1. INTRODUCTION

Web services are based on three key standards: WSDL (Web Service Description Language), UDDI, and SOAP, which enable the description, publication, discovery, and communication between services. However, these standards alone do not provide the necessary automated interoperability required to meet scalability needs and reduce development costs. The first step towards achieving interoperability is through the automatic search for Web services. Once the required Web services are effectively located, other forms of interoperability such as composition, invocation, and monitoring can take place. Therefore, there is a need for a mechanism that allows clients to publish the services they offer and find the services they require in a transparent manner that matches the user query.

Various approaches have been developed to address the problem of Web service search, as presented in [1], [2], [3], [4]. In order to address the issue of intelligently interpreting a query related to the search for a composite Web service, the authors propose an approach that involves the cooperation of two primary agents, which utilize domain ontology. The goal is to ensure that the user is provided with a Web service that satisfies their requirements while being in harmony with all of their constraints. The proposed model facilitates the automatic discovery of Web services and allows for the refinement of the search process, which maps the client query and identifies the various Web services that can satisfy it.

The paper is divided into several sections. The first section provides an overview of general principles related to the search for Web services. The second section presents various approaches for discovering and researching Web services. The third section describes the authors' contribution, which aims to optimize the interpretation of a client's query and facilitate the search for a composite Web service. The fourth section illustrates the performance of the "Matching" algorithm, which helps to effectively select the service that best matches the user's query, using a concrete example. Finally, the paper concludes with some perspectives for further research.

2. GENERAL PRINCIPLES OF THE WEB SERVICE RESEARCH

The research service confronts two actors: the service provider that seeks to announce the best possible services and the user that does not know how to seek the service of his dreams. In this section the researchers will try to answer some frequently asked questions like:

- What is the focus of the research approach?
- Why is the approach used?
- How is the approach conceptualized?

The first question focuses on the object to which we apply the publication and research services, that is to say, the entity that publishes Web services that researchers are looking for, its granularity and visibility.

Two perspectives on services should be included to provide a shared terminology for services: A technologically (IT) and business perspective (business) discussed by Baida et al. in [5]. To these two perspectives, the authors add the goal oriented perspective which reflects the fact that a service exhibits intentionality formulated by the purpose that enables its customers to reach.

A service can be atomic or aggregate [6] and there are two types of specification: the services described as a black box, with some single interface available to other services allowing it to be made; and services described as a white

E-mail address: karima.belmabrouk@univ-usto.dz

box, which includes in its specification information on rules, processes and service structure information are shared with other services [7].
The second question is based on the context of services of research use. This objective means of identification in a service book that directly or indirectly to the needs of the user. These services have been previously modeled and appeared in a service book. Service discovery thus focuses on semantic matching phase between supply and demand services. This mechanism thus emphasizes that refers to operations where the approach found in the directory services that match the query made by the user. This can take many forms such as a specific polling mechanism, direct formulation in the service description language or technical query language; or an expression with a high level language other than the language of service description. The service research generally uses matching techniques (lexical, semantic, ontological, etc. ...). This mapping operation or "mapping" uses a relative similarity according to a given measurement function [8].
The third question concerns the models used as much for the process aspects that for products aspects. On the side of product, we focus on languages that allow users to express and possibly to model their queries when searching services. We identified a few ways:

- Keywords: This is the form used to enter a query in the standards of service directories (such as UDDI).
- Task Model: In this form, the user expresses his request as a job task model. A task is an action performed by an agent or agents to achieve a goal. This is the choice made in the approach proposed by Da Silva Santos in [9].
- Object model: the user expresses his complaint according to a certain structure, where one distinguishes the subject, verb and settings.
- Card Model: The Card is a system of representation to modelize the process in terms of intentional .The graph shows what intentions can be realized through what strategies. This is the choice made in the approach proposed by Mirbel and Crescenzo example in [10].

And we are also interested in the query template, i.e. the form from which the search engine associated with the directory to search. This form may correspond directly to what the user has made in its application, or it is obtained by processing the request to another form that the search engine can process. We identified a few ways:

- Keywords
- XQuery: language optimized for querying all types of XML data.
- SPARQL: This language combines a query language and a protocol to create a Web service in the true sense. Process side, we are interested, first, by the query formulation process. This action concerns the approach with which the end user formula and built a query. We identified three possible values:
- Gross: The formulation process is crude when the user must manually enter the application. This process is completely dependent, and no support is provided by the system.
- Guided: we speak of guided development process when the system has some knowledge about the way in which this process should be conducted. This assumes a prior modeling of this process has been made, and that this modeling is usable by the system. This guide facilitates the formulation of a query and enables to use all the possibilities offered by the model (or language) query.
- Intelligent: The query formulation process is smart when the system is able to reason about the knowledge that is expressed in the request being processed. This assumes that the system relies on the use of ontologies, and these ontologies are used to reason about the concepts appearing in the request and those referenced in the descriptors of services.

Secondly we focus on matching or the "Matching" regarding the mapping between the elements of two sets, in this case those of the application and those of potential results of this query. This process is based on a set of functions to measure the degree of similarity between them. This is achieved by one or more methods of comparison called "matcher"; these are functions used to calculate the distance (lexical, semantic, conceptual, grammatical, etc.) between the two entities. The "Matchers" can combine several techniques in the process of pairing. We identified two possible values to characterize the matching process:

- Lexical: in lexical pairing is textual and syntactic form of the compared elements is preferred. The measurement is made in this case by known metrics such as measuring "Levenshtein" wherever cosine vector, "TF / IDF" method, the difference of method information "Jensen-Shannon."
- Semantics: in the semantic matching, that is the meaning of the compared elements is the basis of the matching process. This type of process presupposes the existence of ontologies for associating hierarchies of concepts to the query items and services to those descriptors. The measure therefore covers a distance in semantic ontology.

The algorithm of "Matching" should allow a search:

- Exact: The results returned by the algorithm must match the definitions of degrees of correspondence explained below for each of the mechanisms.
- Effective: The response time must be more fluid.
- Efficient: All the results should not be too big. Better a small set with a high degree of correspondence.

The algorithm must reduce the number of false positives and false negatives. Moreover, it would be interesting to encourage actors to make an honest description of their service. In other words, if the description is not honest, the service is not bad or included in the results for him.

3. DIFFERENT APPROACHES TO RESEARCH AND DISCOVER WEB SERVICES

Several approaches have been developed in recent years in this field, among all those studied; researchers in [11] gives a comparison between ontology and other techniques on the performance parameters to provide challenges of ontology, in Web service discovery, like the potential conversion of existing non-semantic structure to the semantic descriptions and the less research work in the field of semantic web service that targets to enhance qualitatively or upgrade Web service ontologies or can facilitate the use of ontologies and improve the selection process.

In this paper, authors describe some ones in order to try to form a representative group of the state of the art:

A. *SPOC Approach (for Semantic based Planning Optimized Compositions)*

SPOC was proposed by Daniela Barreiro Claro in her PhD thesis [12]. Specifically, this work focuses on automatic composition of Web services based on the four phases below:

1) Discovery phase;
2) Planification phase;
3) Execution of estimate phase; and
4) Implementation phase.

For the Web service research phase, Claro, D. B. uses a semantic description and offers a domain ontology that organizes Web services using a semantic similarity [12]. Once a task defined, the discovery process researches, in this ontology, Web services that perform it.

B. *GODO Approach (for Goal Oriented Discovery for Semantic Web Services)*

It is a discovery and invocation approach of Web services whose main purpose is to help users to formulate their needs to search semantic Web services such presented by Garcia-Sanchez et al. [13].
The query is expressed in natural language. The approach has a smart editor that helps users to formulate their needs. For this, the GODO approach uses ontologies on the most cited actions that are stored in a referential. Once the user inputs analyzed by GODO, it extracts similar actions in the query of the user and selects the most expressing in the form of goals.
The coupling GODO with a semantic directory as WSMX, DAML-S virtual machine and METEOR-S allows for research. Nevertheless, the goal is not integrated into the semantic descriptor of these directories; it is transformed into keywords by the ”Goal Sender”. GODO considers only the discovery and research services in the design of the system. The publication process and the algorithm of pairing (lexical) are left to the responsibility of the service register. Rather, ontologies are used outside of the directory by the formulation process.

C. *SWASDL-MX Approach*

The acronym SAWSDL-MX stands for Semantic Annotation for WSDL and XML Schema - Machine eXtension. This discovery approach was proposed by Klusch et al. [14]. Its goal is to help users to find and classify the semantic Web services. For this, it used domain ontologies for a hybrid matching based on a combination of lexical and semantic matching mechanisms to provide better performance. Among the hybrid matching the authors cited LARKS and OWLS-MX, which combine formal capacity (based on description logics) and informal capacity for Web services for the Matching process. OWLSMX offers 5 types of matching process, for which one is a purely logical, and other methods are inspired from information search techniques.
The authors claim that an approach based only on lexical matching or semantic one fail because of its limitations. The query expected by the user must conform to the XQuery language. Ontologies are used within the directory to improve search performance.

D. *SATIS Approach*

SATIS is a framework that aims to create Web Service specifications based on the requirements of end-users. This is done in order to implement business processes within a specific application domain.This approach is a semantic Web service discovery approach based on goal. It is developed within the I3S by Mirbel and Crescenzo in [10]. And it is guided by the intentions, it is based on a model to capitalize, reuse and share information and search queries to organize a set of formalized research approaches also reusable and shareable.
This approach is based on Semantic Web techniques and models to propose ways of reasoning and explanation of found Web services, as well as the intentional representation of a process.
The authors propose a model to formalize and reason about the goals and sub-goals. Finally, the proposal allows the reuse and sharing steps (for sub-goals) between research procedures. Specifically, the authors are interested in the construction and use of a queries base making steps of an information search scenario operational and they call it information search process. This knowledge base can be seen as an episodic memory in which the search process is built dynamically, relatively to the context.
In SATIS, three main actors are presented: the designer of Web services, the process modeling expert and the domain expert. And it is based on three ontologies in the same time: the ontology of intentional processes that are annotated with concepts and relations of this ontology enabling sharing and reasoning on these intentional representations, the neuroscientist domain ontology describing medical images and their associated image processing, and the semantic description of Web services given by OWL-S. These queries are represented in SPARQL such as mentioned by Mirbel and Crescenzo in [10].

E. *DRISS Approach*

In this modeling, research and selection Web services approach, the authors propose a requirement centered approach. The user needs are formalized in terms of functional and non-functional requirements, using the formalism of the map specifying the services required

using the intentional service model.
The services research is done by querying the Web service search engine "Service Finder" using keywords extracted from specifications provided by MiS, and the automatic selection of pertinent and high quality services is done by applying the formal concept analysis (FCA) such as proposed by Driss et al. in [15].

F. SECSE Approach

This service discovery approach based on the requirements expressed in natural language was developed by Zachos et al. [16]. Its main function is research and classification of Web services from compound and structured queries based on natural language.
This approach uses an extended directory and it consisted of 4 steps. In the first step, the service query can cut into sentences and then in chips, tagged and modified to include the morphological roots of each term (e.g. travel to travel, and travelers to travel).
In the second step, the algorithm applies procedures to resolve the ambiguity by defining each term to its correct meaning and inserts for each term (for example the definition of a driver is a driver of a plane rather than a pilot of a computer device).
In the third step, the algorithm increases each term with other words that have similar meaning, to increase the probability of a match with a service description (for example driver is a term synonymous with the term conductor which is also included in the query).
In the fourth step, the algorithm makes a "Matching" with all the expanded terms and tagged the query. Which results in a similar set of terms that describe each candidate service expressed using the facet of the service description in the service registry.
SeCSE answers the query in 2 steps: XQuery research, syntactically, to discover a first description sequence of services that meet the research constraints; the information retrieval model of the traditional form of vector space, reinforced by WordNet, refines and evaluates the qualifications of all candidates' services. This approach allows, in two steps, to overcome the limitation XQuery search based on the text.
About WordNet: Online WordNet lexicon plays an important role for the three components of the algorithm. WordNet is a lexical database. First, it divides the lexicon in four categories: nouns, verbs, adjectives and adverbs. The meaning of words, called sense for each class are organized into sets of synonyms that represent concepts, and each synonym is followed by its definition, which contains a phrase defining an optional comment and one or more examples. Second, WordNet is structured according to semantic relations between the meanings of words that link concepts according to Zachos [16].

G. AASDU Approach

It is a multi-agent approach for Web services discovering proposed by Palathingal and Chandra [17]. The system called AASDU (Agent Approach for Service Discovery and Utilization) contains four components:

- A graphical user interface;
- A query analyzer agent;
- A referential system of expertise domains for service agents;
- The services module.

In this system, the user introduces its query as a string of characters using the Graphical User Interface. Then, the query is sent to the Query Analyzer Agent (QAA) which highlights this query by extracting pertinent keywords that will be used to select agent's referential system of expertise domains for service agents.
To do this, the QAA uses a simple variant of the technical TFIDF (Term Frequency Inverse Document Frequency) to highlight the pertinent keywords on the basis of those keywords the QAA agent selects a set of expert agents. Selected agents transmit, subsequently, the parameters of the services with which they are related to the composition agent which invokes one of the services according to user choice such presented by Palathingal and Chandra [17] and Berdjouh [18].

H. SLR Approcah

This approach was proposed by Khani Dehnoi and Araban in [4] who present a systematic literature review conducted on the topic of web services composition. Systematic reviews are intended to assist experts in staying informed about the latest developments in a particular field of research, or to help those seeking information on topics that are currently receiving significant attention, as well as those that have received less attention from researchers, so they can select research subjects more effectively.
The authors present the results of a systematic literature review that examined the current state-of-the-art achievements in web services composition. The study began with a set of research questions, which led to the identification of relevant keywords for searching four well-known digital libraries. The search results were then filtered based on some inclusion and exclusion criteria. The researchers ultimately identified 269 of the most relevant documents, which were analyzed to provide insights into three key areas: bibliography, research quality, and strategy.

I. Summary Table

All the approaches, presented above, can be summarized in the Table I from which we can deduce that among all the cited approaches the one that adopts the multi-agent technology and distributed architecture is the Agent Approach for Service Discovery and Utilization (AASDU) proposed by Palathingal and Chandra [17]. This encouraged us to draw on this approach for our contribution, which will be the subject of the next section.

4. OUR CONTRIBUTION

A. Basic Work

The paper discusses a model that is centered on solving the issue of Web service composition. This model is based on a prototype presented by authors in [19]. The authors recommend that users utilize a graphical interface that provides two types of constraints to enhance the quality of service while executing their queries. They have presented a formal representation of the criteria to make it easier to evaluate the best plan, whether it's a partial

TABLE I. Comparison table for the different approaches

Approach	Operation	Remark
SPOC	Use UDDI service ontology, using semantic Web as Web services description language.	To this day this ontology does not exist - for tests; the author uses OPS 3 (Ontology to Publish Services).
Godo	Query expressed in natural language, intelligent editor, coupling with a semantic directory to enable research.	Its originality is to propose an intelligent process of query formulation independent of the semantic services directory.
SAWSDL-MX	Use of a domain ontology and hybrid matching	More performance guarantee to Web service discovery but not interested in query formulation problem
SATIS	Services are atomic and of Black box type (Card model)	The research process is guided and independent of the directory of semantic services.
DRISS	Use the MiS model because the service is described as an operational perspective and not intentional (WSDL standard) It delegates the search ”ServiceFinder” component by passing the MiS elements as a keywords sequence without semantic.	Its originality resides in the exploitation of card model and MiS model when formulating queries (guided process) and the FCA model when selecting pertinent service.
Woogle	Use of the technique of similarity between operations,	the approach is based on syntactic description of services Web services representation under simple and complete form
SeCSE	Focused on an atomic service of a black box type according to a technological perspective, SeCSE focuses only on the discovery of services and is not interested in publishing process. Proposed matching is just lexical.	The originality of this approach is to improve the process of formulation by the expansion of the query using WordNet. The formulation process is intelligent because it transforms the query expressed in natural language in four successive steps
AASDU	Based on the syntax service descriptions and syntactic comparison between the user query-based keyword, and Web services syntactic descriptions (WSDL).	Adopts the multi-agent technology and distributed architecture and for research it uses the technique ”TFIDF”
SLR	Based on a set of research questions, which led to the identification of relevant keywords for searching four well-known digital libraries. The search results were then filtered based on some inclusion and exclusion criteria.	Identifies 269 of the most relevant documents, which will be analyzed to provide insights into three key areas: bibliography, research quality, and strategy.

or a global plan. Figure 1 illustrates the overall structure of the basic model.

Different types of agents were suggested.

- UserAgent: The cognitive agent ”UserAgent” has the task of handling the user’s preferences and beliefs, as well as synthesizing the constraints to classify them by type.
- ServiceAgent: A group of this type of agents will be automatically created to search for services that meet the user’s request. These agents will filter and rank the services based on their relevance by matching each service with an autonomous agent. These agents are capable of discovering various offers related to the requested service, selecting the best offer, and invoking the service if necessary.
- PlanAgent: The ”PlanAgent” is accountable for the composition of the web services involved in the process. Its aim is to suggest the best possible plan based on the execution time and the specific constraints imposed by the client.

Every ”ServiceAgent” will work together with ”UserAgent” and ”PlanAgent” to suggest a comprehensive plan that can fulfill the user’s request. This is illustrated in Figure 1.

B. The Architecture of the Proposed Model

The Novel proposed model is based on SOA Service-oriented architecture (Service Oriented Architecture). It includes two agents, ”Query Analyzer Agent” and ”Dis-

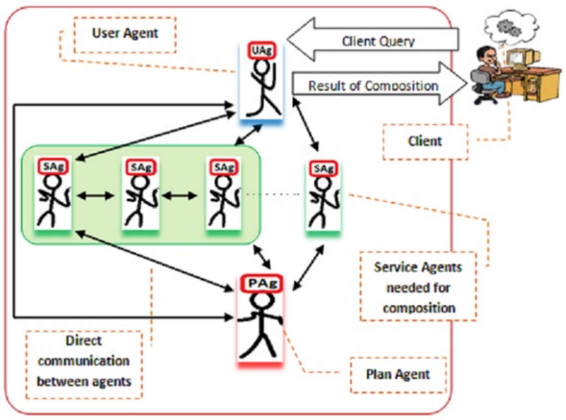

Figure 1. Agent-based Architecture for Automatic Web services composition [19]

covery Agent", a description of Web services registry, and domain ontology.

The proposed model, illustrated in Figure 2, makes it easier to discover services automatically. This is achieved by connecting the client's query with the available services, which is made possible by analyzing and modifying the query through a combination of the Web services registry, domain ontology, and two agents: the "Query Analyzer Agent" and the "Discovery Agent". The model is based on Service-Oriented Architecture (SOA). The prototype's architecture is shown in Figure 2, and it allows for automatic service discovery by analyzing and modifying the client's query to match the available services. The

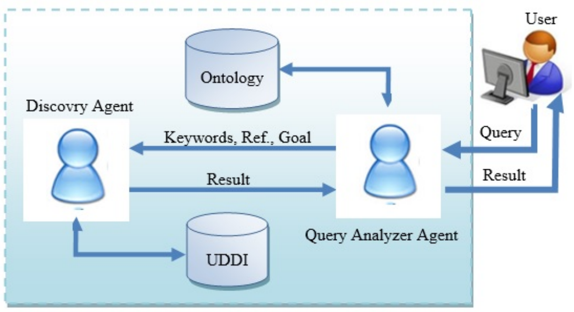

Figure 2. The proposed model architecture.

Query Analyzer Agent (QAA) and the Discovery Agent (DA) are the two primary components of our model:

- Query Analyzer Agent (QAA): The Query Analyzer Agent (QAA) is a responsive agent that acts as an intermediary between the client and the system. It gives the user the option of two types of queries: a guided query using keywords and a free-form query in natural language. The system retrieves the latter and analyses it to extract keywords using an extraction algorithm based on the "Regex" class and a domain ontology.

  The Query Analyzer Agent is responsible for initiating the discovery of web services based on the client's query. It provides the analysis results to the Discovery Agent and ultimately presents the tailored results to the user based on their preferences. The Query Analyzer Agent has three main modules: a communication module for interaction with the user, a query processing module, and a module for communication between agents. Additionally, there is a registry backup module, as shown in Figure 3.

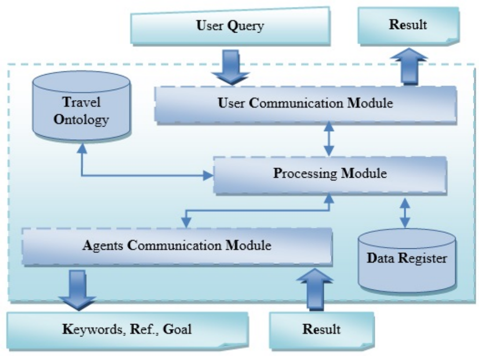

Figure 3. The Query Analyzer Agent Architecture

- Discovery Agent (DA) The Discovery Agent is a cognitive agent responsible for providing semantic descriptions of web services that satisfy the client's query received from the QAA agent. It has an internal architecture that consists of two modules, along with basic services for storing semantic descriptions of web services that satisfy the user query through the UDDI (Universal Description, Discovery and Integration) Web services directory.

  The communication module receives the query from the QAA agent, activates the processing module, and transmits the search results to the QAA agent. The processing module compares the client parameters with the offers of web services, using the Matchmaking algorithm proposed in [20]. This algorithm calculates the degree of correspondence between the functional parameters of web services and those provided by the user, using four main comparison modes: Exact, Plug-in, Subsume, and Fail mode.

  1) Exact mode: In Exact mode, an offer is selected if it precisely matches the user's demand. This means that the inputs and outputs of the user's request are the same as the inputs and outputs of the offer, resulting in an exact match.
  2) Plug-in mode: In the Plug-in mode, an offer is returned if it contains a query where the demand is less than the supply. This means that the demand inputs are a subset of the inputs of the offer and the demand outputs are a superset of the outputs of the offer in the domain ontology. It is an inclusive 'Matching'.
  3) Subsume mode: This mode returns an offer if it is a subset of the query made by the user.
  4) Fail mode: In the Fail mode, the algorithm

returns a false value when there is no match between the user query and the available offers, indicating a failure in matching.

In the "Matchmaking" algorithm used by the Discovery Agent, a score is assigned to each "Matching" mode to determine the degree of correspondence between the functional parameters of Web services and those given by the user. The scores assigned to each mode are: Exact (score = 3), Plug-In (score = 2), Subsume (score = 1), and Fail (score = 0).

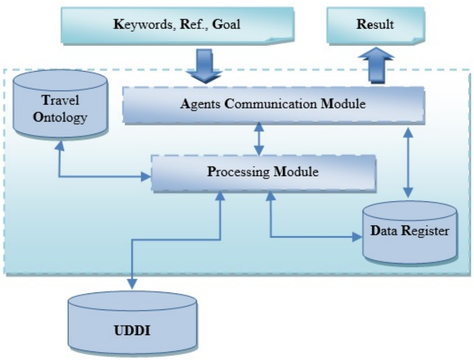

Figure 4. Discovery Agent Architecture

5. ILLUSTRATION AND DISCUSSION

A. *Illustration example*

Imagine that there exist four travel-related web services, namely S1, S2, S3, and S4, that are accessible on the internet. These services possess specific functional characteristics, such as inputs and outputs.

- **S1** having the values (Algiers, Rabat, 20000, 19/12/2022).
- **S2** having the values (Oran, Paris, 22000, 07/12/2022).
- **S3** having the values (Morocco, Tunisia, 23000, 14/12/2022).
- **S4** having the values (Algeria, France, 22000, 07/12/2022).

And suppose a customer looking for a flight from Oran to Paris for 20000 0 DA or less, on 07/12/2022. If we apply the algorithm of "Matching", we will get the following results:

S1:

- Oran to Algiers, mode = Subsume, score = 1
- Paris to Rabat, mode = Fail, score = 0
- 20000 to 20000, mode = Exact, score = 3
- 07/12/2022 to 19/12/2022, mode = Fail, score = 0

Total = 4

S2:

- Oran to Oran, mode = Exact, score = 3
- Paris to Paris, mode = Exact, score = 3
- 20000 to 22000, mode = Fail, score = 0
- 07/12/2022 to 07/12/2022, mode = Exact, score = 3

Total = 9

S3:

- Oran to Morocco, mode = Fail, score = 0
- Paris to Tunisia, mode = Fail, score = 0
- 20000 to 24000, mode = Fail, score = 0
- 07/12/2022 to 07/12/2022, mode = Fail, score = 0

Total = 0

S4:

- Oran to Algeria, mode = Plug-in, score = 2
- Paris to France, mode = Plug-In, score = 2
- 20000 to 22000, mode = Fail, score = 0
- 07/12/2022 to 07/12/2022, mode = exact, score = 3

Total = 7

For the global "Matching", we have:

- S1: score = 6, not bad
- S2: score = 9 Best
- S3: score = 0, Not good
- S4: score = 7 Well

This means that the S2 Web service is identified as the most suitable option to fulfill the client's query, and the agent arranges all the available services in increasing order based on their scores. Finally, any services with a score of zero are excluded from the results.

B. *Discussion of the Client Query Processing Principle:*

At the start of the application, the user has the option to choose between two search modes. The first mode is guided, where the user must enter the necessary information in fields provided on the interface. The second mode is raw, where the user can enter a free-form search query.

1) Case of a guided query: If the user chooses to use the guided search query mode, they need to fill in the required fields on the interface with the relevant information such as the departure and arrival cities or countries, departure date, and price. The Query Analyzer Agent then retrieves the keywords from the input and sends them to the Discovery Agent, which uses the "Matching" algorithm to find the best matches between the user's requirements and the available offers.

2) Case of brut query: This refers to the situation where the user inputs a raw or unguided query in natural language, as shown in Figure 5. The ”Query Analyzer Agent” applies an algorithm to extract relevant keywords from the query and sends them to the ”Discovery Agent” for processing using the ”Matching” algorithm.

After analyzing the client’s query, the Discovery Agent uses the ”Matching” algorithm to find the corresponding services and displays the results. For instance, if the user introduces a query in French language such as ”choisir un voyage entre Oran et Tunis à partir du 12/12/2022 à 25000 DA”, the agent will display the search results in a table, as shown in Figure 5.

Departure	Arrival	Date	flight price	Compagny
Oran	Tunis	12/12/2022	25000	Tunisair
Oran	Tunis	12/12/2022	26000	Air Algerie
Oran	Tunis	30/11/2022	18000	Turkish Airlines
Oran	Monastir	27/11/2022	30000	Air Algerie
Oran	Tabarka	19/11/2022	24000	Tunisair
Oran	Paris	07/12/2022	22000	Air France
Oran	Bahrain	28/11/2022	82000	Gulf Air
Oran	Istanbul	15/11/2022	58000	Turkish Airlines

Figure 5. Brut Query Result

This means that for each query entered by the user, there may be different results returned depending on the type of input.

6. CONCLUSION AND PERSPECTIVES

This paper presents a model developed by the authors to intelligently interpret client queries in a composite Web service search. The model was created using several tools, including Netbeans IDE for the graphical user interface, the Jade platform for agent creation, and ”Protege” for implementing the travel ontology.
Through their studies, the authors discovered multiple methods to accurately fulfill user needs in the Web services search, and they proposed various ideas to assist users in searching for composite Web services both syntactically and semantically. Their approach is based on agents and ontology and models the semantic Web service research. The proposed model includes:

- Query Analyzer Agent (QAA);
- Discovery Agent (DA);
- Travel ontology;
- Register for describing Web services (UDDI).

The authors put the proposed architecture into practice and conducted tests with various user requests related to travel booking to validate it. They achieved results that were in line with the objectives they set and were very satisfactory.

The objective of this paper was to propose a query interpretation tool based on agents. To do this, authors implemented a graphical interface from which the analysis agent will extract the most relevant words by applying a matching algorithm, which will give this agent more power to ensure a better quality of service to the client.
This paper aimed to propose a tool for interpreting queries based on agents. To achieve this, the authors implemented a graphical user interface through which the analysis agent extracts the most relevant keywords using a matching algorithm. This approach enhances the power of the analysis agent and helps to provide a higher quality of service to the client.
In our approach, agents play an important role in the decision-making process. The system is designed to be agent-based, which means that each agent has its own set of rules and objectives, and makes decisions based on its own local information and interactions with other agents.
The agents in the proposed approach are responsible for performing specific tasks, such as collecting and analyzing data, making predictions, and generating recommendations. They communicate with each other through a decentralized network, sharing information and coordinating their actions to achieve the system’s overall objectives.

By using a decentralized network of agents, our approach can be more resilient to failure and more adaptable to changing conditions than a centralized system. Additionally, the use of agents allows for greater flexibility and scalability, as new agents can be added or removed as needed to improve performance or accommodate changes in the system’s requirements.

The authors anticipate the following prospects for their results:

- Researchers can enhance the ”Matching” algorithm by including additional search criteria such as location and service quality to improve accuracy.
- Introduce an ”Indirect Matching” technique in cases where ”Matching” is not possible (composition).
- Expand the model’s capability to solve real-world web services composition issues.
- Develop a generic model that can be scaled up.

References

[1] B. Benatallah, M.-S. Hacid, C. Rey, and F. Toumani, “Request rewriting-based web service discovery,” in *The Semantic Web - ISWC 2003*, D. Fensel, K. Sycara, and J. Mylopoulos, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 242–257.

[2] F. Hadjila, “Composition et interoperation des services web semantiques,” Ph.D. dissertation, Universite Aboubakr Belkaid Tlemcen. Algerie, 2014.

[3] M. Klusch, B. Fries, M. Khalid, and K. Sycara, “Owls-mx: Hybrid owl-s service matchmaking,” in *in Proceedings of 1st International AAAI Fall Symposium on Agents and the Semantic Web, 2005*, 2005.

[4] M. Khani Dehnoi and S. Araban, "A systematic literature review on web services composition," *International Journal of Nonlinear Analysis and Applications*, vol. 13, no. 1, pp. 2821–2855, 2022.

[5] Z. Baida, J. Gordijn, and B. Omelayenko, "A shared service terminology for online service provisioning," in *Proceedings of the 6th International Conference on Electronic Commerce*, ser. ICEC '04. New York, NY, USA: Association for Computing Machinery, 2004, p. 1–10.

[6] C. Rolland and R. S. Kaabi, *Designing Service-Based Coorperative Systems*. IGI Global, 2005.

[7] N. Arni-Bloch, J. Ralyté, and M. Léonard, "Service–driven information systems evolution: Handling integrity constraints consistency," in *The Practice of Enterprise Modeling*, A. Persson and J. Stirna, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 191–206.

[8] M. Klein, B. K. Ries, and M. Mussig, "What is needed for semantic service descriptions? a proposal for suitable language constructs," *International Journal of Web and Grid Services*, vol. 1, p. 328, 2005.

[9] L. O. B. da Silva Santos, G. Guizzardi, L. F. Pires, and M. van Sinderen, "From user goals to service discovery and composition," in *Advances in Conceptual Modeling - Challenging Perspectives*, C. A. Heuser and G. Pernul, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 265–274.

[10] I. Mirbel and P. Crescenzo, "From end-user's requirements to web services retrieval: A semantic and intention-driven approach," in *Exploring Services Science - First International Conference,IESS*, ser. Lecture Notes in Business Information Processing, J. Morin, J. Ralyté, and M. Snene, Eds., vol. 53. Springer, 2010, pp. 30–44.

[11] M. Fariss, N. El Allali, H. Asaidi, and M. Bellouki, "Review of ontology based approaches for web service discovery," in *Smart Data and Computational Intelligence*, F. Khoukhi, M. Bahaj, and M. Ezziyyani, Eds. Cham: Springer International Publishing, 2019, pp. 78–87.

[12] D. B. CLARO, "Spoc: Un canevas pour la composition automatique de services web de la realisation de devis," Ph.D. dissertation, Universite d'Angers, 2006.

[13] J. M. Gomez, M. Rico, F. Garcia-Sanchez, I. Toma, and S.-K. Han, "Godo: Goal oriented discovery for semantic web services," in *5th International Semantic Web Conference*, 2006.

[14] M. Klusch, P. Kapahnke, and I. Zinnikus, "Sawsdl-mx2: A machine-learning approach for integrating semantic web service matchmaking variants," in *IEEE International Conference on Web Services*. IEEE, 7 2009, pp. 335–342, doi 10.1109/ICWS.2009.76, isbn 978-0-7695-3709-2.

[15] P. P. Maglio, M. Weske, J. Yang, and M. Fantinato, *Service-Oriented Computing*. Springer Berlin Heidelberg, 2010, vol. 6470.

[16] K. Zachos, N. Maiden, X. Zhu, and S. Jones, "Discovering web services to specify more complete system requirements," in *In Proceedings of the 19th international conference on Advanced Information Systems Engineering Lecture Notes in Computer Science*, J. Krogstie, A. Opdahl, and G. Sindre, Eds., 2007, pp. 142–157.

[17] P. Palathingal and S. Chandra, "Agent approach for service discovery and utilization," in *37th Annual Hawaii International Conference on System Sciences*. IEEE, 2004, p. 9 pp.

[18] C. Berdjouh, "Une approche basée agent pour la decouverte de services web," Master's thesis, Université Kasdi Merbah, Ouargla, Algérie, 2009.

[19] K. Belmabrouk, F. Bendella, and M. Bouzid, "Multi-agent based model for web service composition," *International Journal of Advanced Computer Science and Applications*, vol. 7, 2016.

[20] K. Belmabrouk, "Modèle multi agents pour la composition des services web, fondé sur des techniques de planification," Ph.D. dissertation, Université des Sciences et de la Technologie d'Oran - Mohamed Boudiaf USTO-MB, 2017.

Karima Belmabrouk received the engineering diploma in computer science in 1998 and her magister degree in 2006 from the University of Science and Technology of Oran, USTO-MB as well as her PhD degree in 2017. Since December 2006, she is an assistant professor and researcher in the computer science department of the USTO-MB. Her research interests are based on agent-based modeling and simulation, Web services composition, ontology and semantic Web as well as planning in artificial intelligence.

Latifa Dekhici is currently working as an associate professor in the faculty of mathematics of computer science at the University of Sciences and the technology of Oran (USTO- MB) and also a member of the LDREI laboratory in the High School of electrical and electronic engineering of Oran. She has received her engineering diploma in computer Sciences in 2002, then her magister degree in artificial intelligence and pattern recognition in 2005 and she obtained a PhD degree from the same university. Her thesis was about operating theatres modelling and surgeries scheduling. Dekhici contributed to many combinatorial optimization fields as power dispatch, power system design, healthcare scheduling and green vehicles routing. She presented new versions of bio-inspired metaheuristics in several international journal papers and in 3 books.