International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-9 Issue-3, February 2020

Addressing Natural Language Request Complexity: At Semantic Web Service Discovery Horizon

Aradhana Negi, Parminder Kaur

Abstract*: The wondrous influence of semantic web on Service-Oriented Architecture pushes it towards a realistic and self-driven architecture where publication, discovery, selection, composition, and monitoring of services are semi-automatically performed on the behalf of their hosts or mediators. In the direction of this realistic and self-driven architecture, this research work is adding one more realistic aspect of 'interpretation of natural language request' to making the service discovery more usable for novice users. Three contributions have been made: (1) description of natural language request using six-slab range (2) two algorithms for extraction, sub-request generation, inclusion of semantics and semantic matchmaking of natural language request, and (3) evaluation of proposed strategy with two semantic formalisms. The proposed algorithms handle each complex service request as an individual entity and extract the demand/s of the request by decomposing it to the simple request from conjunction, condition, and negation-oriented natural language request. The experimental evaluation of the proposed strategy signifies the given algorithms. The proposed work and result evaluation is a part of on-going research on a generic discovery mechanism for semantic web services.

Keywords: Composite Services, Natural Language Processing (NLP), Natural Language Request (NLR), Semantic Web Service (SWS), SWS Discovery

I. INTRODUCTION

In the distributed software systems, the discovery of the desired service is a fundamental activity in the entire spectrum of service-related tasks. Service discovery refers to a process to retrieve desired services by matching service request with available potential service descriptions [1]. A service description is a document provided by service provider to convey the functional and non-functional specifications of the service at the time of service discovery. The service description is advertised by service provider to make it discoverable and if matched then invocable for service requester. The service request is the another essential part of service discovery. It is a set of requirements given by the service requester for discovering the relevant service/s. The discovered results must satisfy the service requester for the total success of any service discovery system. The linkage of concepts in a service description document with some ontology, makes the web services - the Semantic Web Services (SWSs). Ontology defines the concepts and represents an area of domain-specific knowledge [2]. Here, the basic aim of service discovery process is to first understand and then effectively match the service request with the service description to find desired services.

Revised Manuscript Received on February 05, 2020.
* Correspondence Author
Aradhana Negi*, Ph.D. Research Scholar, Department of Computer Science, Guru Nanak Dev University, India.
Parminder Kaur, Assistant Professor in the Department of Computer Science, Guru Nanak Dev University, India.

The large number of available published web services necessities the provision of automated on-the-fly discovery, matchmaking and composition [3]. However, the full automated mechanisms for service discovery [4] can be achieved only when the system automatically understands the user requirements with higher accuracy. The significant amount of research work [3][5][6][7][8][9][10] considered the automatic or semi-automatic discovery and selection of suitable web service as a challenging task. The existing research work on service discovery can be broadly categorized as keywords-based, mediator-based, semantic-based, semantic tagging-based, and ontology-based matching search [11]. Though, all these categories of service discovery claims for fine results but they ignore the fact that ultimate result of service discovery is for users and in present era, the 'better user experience strategy' of business is the core drive for every business organization.

Moving towards more advanced technologies, the trend in Computer applications is shifting to Natural Language Processing (NLP) for example, Amazon Echo [1] provides voice-controlled smart personal assistant service '*Alexa*' to their users. The '*Alexa*' can capture information from services like weather, music, telecommunication, news etc. Some other technology giants such as Apple, Google, and Microsoft are also working for voice-control base assistants. This sort of advancement of involvement of NLP in service discovery is the need of the hour [12][13].

Usage of NLP techniques in service discovery is important from various perspectives such as **(1)** ontologies have natural language, therefore NLP techniques assist in better understanding the context of web services [14], **(2)** natural language interfaces are user-friendly and convenient for novice users due to the features of hiding internal complexity and making the users free to have intimate knowledge of *SWSs* [15][16], and **(3)** the textual part of the services description is equally significant as functional and non-functional specification of that service description [17].
The prime aim of this research work is to discover suitable SWSs using natural language based user requests.
Natural Language Request (NLR) as an input for any

[1] Developed and launched by amazon.com in 2014

discovery system has two sides. At one side, it is natural for users

because the system hides the execution complexity of service requests from service users [18] while at another side, it creates challenges such as coding the natural language, inferences from codified knowledge, automatic rule

digestion, and providing single solution for ambiguity resolution [19] for the system developers.

Although a service request can also be given using keyword-based or form-based approaches but literature proves that "posing natural language queries are more precise than the keyword approach and at the same time they are more natural than the form-based approach" [20]. The form-based search systems "restrict user for understanding and selecting the fields and tags" [21] while the keyword-based search systems is "insufficient to discover services whose functionalities are similar to a query" [10] and also "lack clear specification of the relations among words" [22].

As a lot of work has been done on NLR but to best of our knowledge none of the contribution has discussed the connectives with positive and negative orientation for NLR. The proposed work addresses NLR for SWSs and present the contributions as: **1)** describing the complexity of NLR using simple, conjunctional, conditional connectives with positive or negative orientation in the form of Six-Slab range, **2)** the algorithms for handling NLR, adding semantics to NLRs and semantic matchmaking of NLRs and web services, **3)** implementation of proposed strategy with two semantic formalisms i.e. OWLS-TC V_3 [23] and WSMO [14]. Here, semantic formalism means the format which semantically describes a web service description document. Currently, there are three pure semantic formalisms for SWSs i.e. **1)** Web Ontology Language-Semantics (OWL-S), **2)** Web Service Modelling Ontology (WSMO) and **3)** WSMO-Lite. The research work is limited to two semantic formalisms due to the availability their web services. With the indented vision, the paper is organized as VI Sections. Section II discusses the nature of problem. Section III briefly presents the related work on NLR for service discovery and composition. Section IV presents the proposed work for NLR. In Section V, the experimentation, respective results and comparison with existing approaches are given. Lastly, Section VI presents the concluding remarks of paper.

II. PROBLEM DOMAIN

NLR is a request for discovering single service or set of composite services using NLP techniques. A NLR may contain more than one sentence block and further a sentence block may correspond to single service. Therefore, composition of multiple services is also a concern of this paper. The paper mainly explores the relationships among NLR elements i.e. verbs, nouns, or adjectives [24] used to find and compose the suitable web services. Five types of connectives have been addressed for NLRs i.e. **1)** simple, **2)** conjunction-oriented, **3)** conditional-oriented, **4)** negation-based, and **5)** mixed. In this paper, the word '*complexity*' means the state of having more connectives or nested connectives in NLR. In English language, a sentence can be joined as a whole or divided into parts using the conjunctions. The conjunctions are mainly of three types: **1)** coordinating, **2)** subordinating, and **3)** correlative. Coordinating conjunction e.g. and, or, but, so, also, moreover etc. joins two independent sentences/phrases of equal significance. In subordinating conjunction, one part is dependent on the independent part of the sentence e.g. while, if, when, next, then, because, until etc. Correlative conjunctions are the conjunctions that appear together as pair e.g. either/or, neither/nor, rather/then, whether/or etc. The conditional connectives are subordinating conjunctions by nature, where one part expresses condition/s to be fulfilled and another part expresses the consequence/s of fulfilled condition/s. The conditional connectives are if, when, else, even if, only if, in case, as long as, until, unless etc. A conditional sentence can appear with one or more conditional connectives. According to Narayanan, Liu and Choudhary [25], the maximum conditional sentences can be presented by using subordinating conjunction ***If***.

Further, negation is a unary connective under logical connectives which excludes the semantic value of a particular condition/subject/object from the given statement. Negation

Table- I: Examples of types of NLR

NLR Type	NLR Example
S +ve	Find addresses of pediatric hospitals in city
S -ve	Book appointment in pediatric hospital but not on Saturday
C^2 +ve	Find Neuro hospital or clinic in city
C^2 -ve	Find job for computer operator with no contract
C^2 +ve	If movie XYZ available then book 3 tickets else reserve a table in Thai restaurant
C^2 -ve	If nearby restaurant then book a table else book movie tickets but not premium tickets
C^3 +ve	If weather is clean then if adventure activities searched in city and book 7 seated car from XX to YY else search on-demand Hollywood films with HD feature on XYZ platform
C^3 -ve	If weather is clean in Paris then if flight ticket from TT to Paris is available on 18 November then reserve 3 business class seats and book cab from Paris Airport to Rouen else if I can find holiday package in Singapore then book 2 flight ticket to Singapore and reserve 3 star hotel not nearby 7 km of Airport.

is an influential connective that can affect the meaning of whole sentence. An obvious question is whether the negation is significant for discovering services? The answer of this question is *yes* because the negation-oriented requests "are a very usual and expected input from the user" [26][27] and also these "provides great textual and contextual significance at a particular point of discourse" [28]. Here, negation is not taken as condition but considered as orientation which can change the direction of results for the discovery of servicesThe connectives in NLR represent request of composite services as the sentence blocks. For composing services, the request should be divided into parts and each part must be semantically searched from service store. The example of each type of considered NLR along with orientation is presented in Table I.

In Table I, 'S +ve' denotes simple positive request while 'S –ve' is simple negation-oriented request. Same notation 'C^{2}' is used for conjunction and conditional connective. In case of 'C^{2}', the request is divided up to the extent of S +ve or S –ve. The nested conjunction and conditional requests are denoted with 'C^{3}'. Each type of NLR is refined to get noun, verb, adjective.

III. RELATED WORK

This section briefly describes the related work for NLR in service discovery, composition, selection, and query generation. The related work, organized in chronological order, has been discussed in Table II with five attributes describing the 'year', the 'approach' used, 'active zone' of research, 'evaluation' for evaluated service requests and, 'potential' of approach. The triple cross sign (×××) depicts no evidence of evaluation i.e. only conceptual models have been discussed. From Table II, it has observed that the total number of NLRs taken for evaluation varies extensively. The five

Table- II: Attribute-based analysis of related work

Ref.	Year	Approach	Active Zone	Evaluation	Potential
[29]	2006	Coupling semantic and ontological info with natural language constructs	D, dC	1 sample sentence	Mapping extracted abstract composition to service operations for fulfilling On-demand service composition
[30]	2006	Transformation of NLR to Predicate Calculus	Qg	×××	*Near Solution* for under-constrained and over-constrained NLR
[31]	2006	Transformation of NLR to Semantic Object Behavior Language (SOBL)	D	×××	Reference model usage for sticking different services together
[32]	2006	NL-based Storybooks from business experts are mapped to semantic of application logic.	W	×××	Involvement of business experts in defining business processes using NL text
[1]	2007	Automated Question-Answering with Semantic web Services	D	×××	Building data sources for automatic answering system
[33]	2008	QuestIO system :Natural Language Interface for structured information	Qg	36 question on GATE(659) & Travel (3194) knowledge base	Exploitation of structure of ontologies, fuzzy string matching, and ontology similarity metrics.
[16]	2009	Natural language interface to Web services	C	127 Self-generated queries on 312 self-generated WS	Extraction of abstract flow from NLR
[34]	2009	Representing NL queries as semantic network	Qg	12 Self-generated sentences on proposed system	Mapping a sentence to sub-network of concepts of a larger ontology
[35] [36] [37]	2009 2010	On-demand service composition using unrestricted NLR	dC	1 NL phrase on WComp middleware	Conceptual distance between user request and service configuration and Usage of aspect-oriented template
[15]	2011	Automatic SWS discovery using NLP techniques	D	×××	Technique for computing semantic distance between ontological concepts
[38]	2012	Content based Service Discovery using WordNet	D	10 Text content data on OWL-S TC V4	Comparison with keyword-based search approach on same dataset.
[14]	2013	SWS discovery using NLP techniques	D	61 self-generated queries on 35 self-generated WSMO based services	Combination of Level matching, Jaccard matching and similarity matching Algorithms
[39]	2013	Computational model to capture cognitive abstract for facilitating query cognitive canonicalization	Qg	243-TREC, 1365-MSQA, 3400-web search engine queries on DBpedia	Usage of semantic strata of cognitive psychology to extract semantic sub net from query
[20]	2013	Semantic Web Search using Natural Language	D	68 Self-generated queries on an accommodation ontology transformed from hotelypenziony.cz	Geospatial search, NL in Czech language, statistical semantic model for semantic analysis of NLR
[40]	2017	Handling conjunctions in NLR and dependencies between services	D,C	80 Self-generated queries on OWL-S TC V4	Service interface graph for service dependency realization

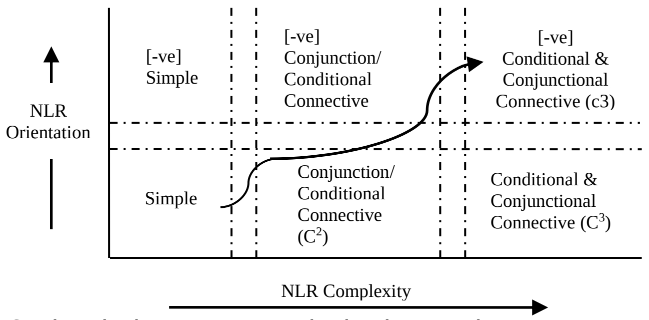
research articles [1][15][30][31]32] have no empirical evaluation while other research article [14][16][20][29][33][34][35][36][37] [38][39][40] have used the benchmarks for validation. The notations 'D' for discovery, 'C' for composition, 'dC' for dynamic composition, 'Qg' for Query generation and 'W' as an exception for Co-existence of NL and semantic web.

IV. PROPOSED WORK FOR NLR

The mainstay of the current research work is to propose a generic discovery mechanism for SWSs. For realizing the objective of handling NLRs, the proposed strategy is divided into four parts: **1)** NLR Six-Slab range, **2)** handling complex NLR, **3)** inclusion of semantics, and **4)** matchmaking of NLRs with web services. The next four sub-sections will explain the respective parts.

A. NLR Six-Slab Range

The connectives discussed in Section II, are dimensionally depicted as per Six–Slab range as given in Fig. 1. In the cartography shown in Fig. 1, the horizontal dimension depicts the increasing complexity of service request whereas



Curved arrow line depiction: Negation oriented conditional conjunctional NLR *

Fig. 1.Six-Slab range for NLR

B. Handling Complex NLR

To discover relevant services, the simple service requests are only passed through pre-processing and negation handling scheme to fetch final semantics of NLR while the complex service requests are decomposed to get simple request from each decomposed part. As NLR does not have explicit input, output and other conditions, so the text of each NLR has

Input: Natural Language based Request (NLR)
Output: Concept, PoS, Synonyms, Sense of each token in NLR

Algorithm Begin

1. *Sub-Request =* Ø
2. *Cond =* Ø
3. *Conj =* Ø
4. ***If** (Cond =* Ø *&& Conj =*Ø *in NLR)* *// skipping in case of simple or simple negation NLR*
5. *Jump to Step 19*
6. ***Else If** (conditional connective in NLR) then do:* *//fetch condition or condition-conjunction*
7. *Increment Cond*
8. *Collect NLR to Sub-Request*
9. ***For each** Sub-Request*
10. ***If** (conjunctional connective) then do:*
11. *Increment Conj*
12. *Collect Sub-Request to Sub-Sub-Request*
13. ***End For***
14. ***Else If** (conjunctional connective in NLR) then do:* *//fetch conjunction*
15. *Increment Conj*
16. *Collect NLR to Sub-Request*
17. ***For Each** Sub-Request/ Sub-Sub- Request do:*
18. ***If** (Negation) do:*
19. *Note index of Negation and get noun or verb restricted by Negation*
20. *Pre-Processing* *//Perform PoS tagging, Stemming and Stop-word removal*
21. ***Call Algorithm II*** *//Semantic inclusion*

Algorithm I. NLR extraction and sub-request generation process

the vertical dimension expresses the change in orientation of service request. The line heading towards extreme right corner describes the most complex type NLR where conjunction, condition and negation connectives come together in a single request i.e. C^3 –ve (see in Table I). The Six-Slab range can express any service request with the connectives discuss above. The examples, experimentation, and interpretation presented in this paper follows the blueprint of the Six-Slab range.

analysed with NLP techniques. Instead of using any natural language processor, the 'Algorithm 1' has been used to exploit the given NLR. Unlike existing approach to get Part-of-Speech (PoS) tags first, the proposed approach stabs to find connectives in NLR first.

Step: 1 - Decomposition with conditional connectives

|| *If weather is clean and adventure activities searched in city* (**sub-request [1]**)
|| *then book 7 seated car to YY* (**sub-request [2]: (dependent**)
|| *else search on-demand Hollywood social movie on XYZ* (**sub-request[3]**)

Step: 2 - Decomposition with conjunctional connectives

|| *and adventure activities search in city* (**sub-sub- request [1]**)
|| *then book 7 seated car to YY* (**sub- request [2]: (dependent**)
|| *else search on-demand Hollywood social movie on XYZ* (**sub- request [3]**)

Step: 3 - Pre-processing

|| *(If) weather (noun) clean (adjective)* (**sub- request [1]**)
|| *(and) adventure (noun) activity (verb) search (verb) city (noun)* (**sub-sub- request [1]**)
|| *(else) search (verb) on-demand (verb) Hollywood (noun) social (adjective) movie (noun) on XYZ* (**sub- request [3]**)

Fig. 2 Decomposition of example NLR using Algorithm. I

After decomposing the NLR, the algorithm adds the sub-part to sub-request list. From literature, it has observed that in most cases, the conditional NLRs have coordinated conjunctions incorporated in it whereas service request with coordinated conjunction do not comprise conditional connectives commonly. Henceforth, the conditional NLR incorporated with conjunctions and negation has been considered as the most complex queries in the implementation phase. The 'If-Then-Else', 'While', 'When', 'And', 'Or', 'But', 'Also', and 'No', 'Not' are taken as possible conditional, conjunctional and negation connectives respectively. When a service request is given then the system captures indexes of connectives and break down the given request into sub-requests. Next, each of the sub- request is checked for conjunction connective until no further decomposition is possible and this process result with collection of simple requests. In case of no conditional connective found, the NLR is still checked for conjunctions. Once a NLR is decomposed, the each of its fragment is looked for negation to decide whether the fragment is a simple request or simple request with negation. If 'No or 'Not' is investigated in request, then the main verb or noun associated with negation is bagged to exclude the particular part from results and further request is processed as a simple request. After completing the decomposition process as per Step 5-19 in Algorithm I, all sub-parts of service request are tagged with CLAWS PoS tagger to get noun, verb and adjective. Next, the Porter stemmer algorithm is applied to stem the words to generate terms/concepts as given in Step 20 of Algorithm I. For handling abbreviations e.g. 'ISBN' and badly asked service request e.g. 'find adventureactivities in city', the algorithms proposed by [12] have been employed to resolve the raised conflicts. The result of preprocessing tasks next fed to Algorithm II which search the obtained concepts through ontologies to include respective semantics. An example "If weather is clean and adventure activities searched in city then book 7 seated car to YY else search on-demand Hollywood social movie on XYZ" given in Fig. 2 showcases the result of first twenty steps in Algorithm

Input: (1) Concepts and PoS extracted with Algorithm 1
(2) OnTo
(3) WordNet semantic network
Output: Return semantic information of concepts

Algorithm Begin
1. **For Each** *Concept*
2. Look up WordNet with {*Concept, PoS*}
3. Get corresponding synset, sense and sense number
4. If(*Concept* ϵ OnTo)
5. Capture superclass and subclass concepts from OnTo
6. Note the list of matched ontology
7. **End For**
8. Return {Concept(including Ontological concepts, if found), PoS, synset, sense, sense_number}

Algorithm End

Algorithm II. Inclusion of Semantics

C. Inclusion of Semantics

A free-form request is just a stream of characters [16] and to interpret it rightly, it must be analyzed with some referenced ontology. Ontology represents a ceremonial way to describe the linked concepts of a specific domain. In ontology, the object/entity, classes, class attributes, relationship between different objects and implied restrictions are defined as per domain requirement for example BioPortal (https://bioportal.bioontology.org/) is a repository of 716 medical Ontologies and 9,419,848 medical classes. In Algorithm II, two categories of ontologies i.e. WordNet [41] and service ontologies (OWL-S TC V_3 and WSMO service ontologies) are considered to add semantic information to the extracted concepts of NLR.

Table-III: Terminology for Algorithm III

Notation	Description

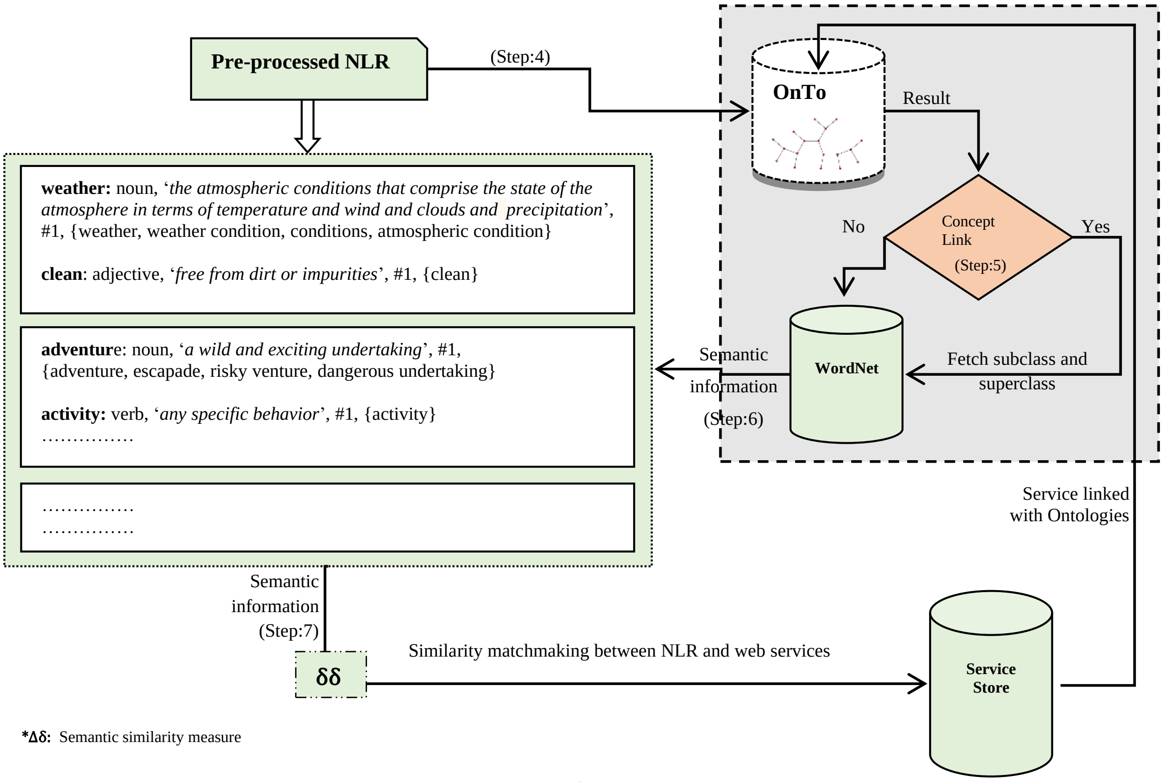


Fig. 3 Capturing semantics from WordNet and ontologies

δδ	Overall semantic similarity measure for NLR and web services
ńδ	Noun similarity
úδ	Verb similarity
áδ	Adjective similarity
śδ	Sense similarity
α, β, γ	Weight for noun, verb and adjective in NLR.
S^d	Semantic distance
W	Associated link strength of shortest path connecting two concepts
IsA	Is a relationship
HasA	Has a relationship
C(p)	Number of children nodes
D(p)	Depth of parent node in hierarchy
R_{Nc_i}, S_{Nc_j}	Noun input from request (NLR) and web services
R_{Vc_i}, S_{Vc_j}	Verb input from request (NLR) and web services
R_{Ac_i}, S_{Ac_j}	Adjective input from request (NLR) and web services
S_{Rc_i}, S_{Sc_j}	Sense input from request (NLR) and web services
Rel	Adjective relations from WordNet
Ant	Antonym adjective relation
Sim	Similar adjective relation
Pf	Penalty factor

WordNet is a semantic network for English language. It expresses the short definitions of concept, set of synonyms for concepts and some other relationships (not considered in the scope of this paper). Prerequisite part for inclusion of semantics is that all the imported ontologies of services must organized as noun, verb and adjective ontologies into a database called OnTo. Here, OnTo database is not alternative for ontologies but provide support [5] to find ontologies for NLR. OnTo infers that a NLR matching with service ontologies may match best with the source service of imported ontology [17]. OWL DL reasoner is used to capture superclass and subclass of matched concept with subsumption relationships i.e. exact, plugin etc. and further the result is added in list of a concepts.

Additionally, every concept is searched in WordNet 0ntology. The WordNet mainly links nouns, verbs, adjectives, and adverbs by means of semiotic associations. It consist three separate storages: one each for nouns and verbs and a third one for adjectives and adverbs. Java WordNet Library (JWNL) [42] and Java WordNet Interface (JWI) [43] are used to obtain sense, sense number and synset of a concept. The reason for retrieving sense of a concept is the implementation of a novel notion i.e. '*sense involvement'* in semantic similarity calculation as given in [44].

D. Matchmaking Process

One of the most important steps in service discovery is the matchmaking process of user request with existing service descriptions for fetching the best matched service. The matchmaking process uses the

similarity measure given in [44]. The adopted semantic similarity metric uses a combination of two well established similarity metrics i.e. Extended Gloss Overlapping Method (EGOM) [45] and Similarity Integrating Multiple Conceptual Relationships (SIMCR) [46] along with the notion of sense inclusion as given in Algorithm III.

Input: (1) Concept, PoS, Synonyms, Sense of a concept from NLR
(2) Web Services (WS)
Output: Semantic Similarity measure ($\delta\delta$)

Algorithm Begin

1. $\alpha, \beta, \gamma =$ average of total extracted nouns, verbs and adjectives
2. **For** (nouns, verbs and adjectives in WordNet) do:
3. C(p) = number of children nodes
4. D(p) = depth of the parent node
5. $W_{Is\,A}(e) = \frac{1}{2+\log C(p)} * e^{-\log(D(p)+1)}$
6. $W_{Has\,A}(e) = 1 - \left[\frac{1}{2+\log C(p)}\right] * e^{-\log(D(p)+1)}$
7. $S^{d}\left(\mathbf{R}_{C_i}, \mathbf{S}_{C_j}\right) = \sum W_{Is\,A}(e_i) + \sum W_{Has\,A}(e_j)$
8. $\acute{n}\delta / \acute{v}\delta\left(\mathbf{R}_{C_i}, \mathbf{S}_{C_j}\right) = e^{-S^{d}}$
9. For each (adjective in WordNet) do:
10. **If** ($0 \le rel \le 1/(Pf+1)$) then do:
11. $\acute{a}\delta\left(\mathbf{R}_{Ac_i}, \mathbf{S}_{Ac_j}\right) = \frac{\cos(A_1, A_2)}{pf+1}$
12. **Else**
13. $\acute{a}\delta\left(\mathbf{R}_{Ac_i}, \mathbf{S}_{Ac_j}\right) = \cos\left(\mathbf{R}_{A_{c_i}}, \mathbf{S}_{A_{c_j}}\right)$
14. $\acute{s}\delta\left(S_{R_{c_i}}, S_{S_{c_j}}\right) = score\left[S_{R_{c_i}} \cap S_{S_{c_j}}\right]$
15. $\delta\delta(\mathrm{NLR}, \mathrm{WS}) = \sum\left[\alpha * \acute{n}\delta\left(\mathbf{R}_{Nc_i}, \mathbf{S}_{Nc_j}\right) + \beta * \acute{v}\delta\left(\mathbf{R}_{Vc_i}, \mathbf{S}_{Vc_j}\right) + \gamma * \acute{a}\delta\left(\mathbf{R}_{Ac_i}, \mathbf{S}_{Ac_j}\right) + \acute{s}\delta\left(\mathbf{S}_{R_{c_i}}, \mathbf{S}_{S_{c_j}}\right)\right]$
16. Return ($\delta\delta$)
17. **Algorithm End**

Algorithm III. Matchmaking for Service Request

The terminology for Algorithm III is presented in Table III. SIMCR uses similarity measure over three ontological relationships i.e. Is-A, Has-A and Antonomy and outperform the other existing similarity measures. Is A and Has A relationship define semantic inheritance and semantic object composition, respectively [46]. Similarly, EGOM utilizes the brief description of the concepts given in WordNet. Noun and verb mainly hold hypernym, hyponym, meronym, holonym, and troponym semantic relation in WordNet. As noun-noun and verb-verb association has similar relations between them so a common strategy is used for their similarity computation i.e. $\acute{n}\delta/\acute{v}\delta\left(\mathbf{R}_{c_i}, \mathbf{S}_{c_j}\right)$. The notion of negative exponential semantic distance ($-S^{d}$) in Algorithm III has employed with the fact that the lesser semantic distance (S^{d}) leads to the greater semantic similarity. The semantic distance ($-S^{d}$) is a weighted-edge measure that used to assess the shortest semantic path between two given concepts. The edge weight relies on two attributes i.e. number of children $C_{(p)}$ and depth $D_{(p)}$of the parent node in the hierarchy [15]. The adjectives hold five relations: similar to, attribute, see also, antonym and derivationally used form in WordNet but in many cases these five relations do not come together for an adjective. Out of these five relations, Antonym (ant), is the chief one which organizes two opposite nature of nouns through adjectives whereas the other relations explicitly form a set because of the some similarity or relatedness of their pOs (part-of-speech) terms. For example, 'Simple' is the antonym of 'Complex' whereas 'Complexity', 'Complicated', 'Compound', 'Complexness' form a group with related features. The sense describes the meaning of noun, verb and adjective in the form of a short definition called gloss. Sense similarity focuses on gloss-based semantic computation by exploiting glosses for the measurement of semantic relatedness. The absolute sense similarity score is normalized between 0 and 1, by divided it with the summation of all tokens presented in the participating senses from both the user request and the service description document.

V. IMPLEMENTATION ANALYSIS

This section provides details about experimentation done for the proposed work. The system with 8th Generation Intel Core i5 4 cores desktop having Windows 10 Pro, 8GB DDR4 RAM and 1TB ATA hard drive has been used for experimentation. The environment used for performing experimentation is Java.

The publicly available OWL-S TC V_3 and extended private WSMO dataset have been used to search services against given NLR. The OWL-S TC V_3 has 7 categories, 1007 services, 32 ontologies while WSMO dataset has 14 categories, 35 services and 14 ontologies. We have extended WSMO dataset by adding 3 categories, 4 ontologies and 15 services i.e. cart Service (5), postal code & geography (5), food finder (5).

Table-IV: NLR Distribution aligned with Six-Slab range

Complexity	Orientation	NLR Distribution	
		OWL-S	WSMO
Simple (S)	+ve	30	14
	-ve	40	20
Conjunction/Conditional Connective (C^2)	+ve	30	18
	-ve	40	20
Conditional Coordinated Conjunction (C^3)	+ve	35	22
	-ve	45	25

Both the taken datasets are completely different in terms of size and coverage, thus the experimentation levels are also different for both the datasets.

In case of OWL-TC V_3, we have taken 220 NLR however, for WSMO, the total 119 NLRs have been used for evaluating the proposed algorithms (see Table-IV). The NLRs for OWL-S TC V_3 have been generated from the 29 test queries given along with the dataset. The NLRs for WSMO dataset are inspired from Sanger et al. [15]. The NLRs are extended with negative and positive orientation and also these are aligned with Six-Slab range described in Section II.

At initial level, the input service requests are examined for sub-request using Algorithm I and generate data in the form of concepts. To fulfil prerequisite condition for experimentation, this paper has worked on 32 ontologies from OWL-S dataset and 18 ontologies from WSMO dataset to divide them manually in object, action and attribution ontologies for the nouns, verbs and adjectives respectively. To best of knowledge, this is one of its kind evaluations performed on two semantic formalisms.

Instead of testing all service requests together with entire dataset once, the four test cases for OWL-S TC V_3 and two test cases for WSMO have been performed. The size of WSMO dataset is very small so only two test cases have been performed. The four test cases on OWL-S TC V_3 are formed by taking different percentage of NLRs and datasets i.e. 30%, 50%, 70% and 100%. The two test cases on WSMO are evaluated using 50% and 100% of NLRs and available WSMO web services. The analysis of experiments has been done using precision, recall and F-measure/score metric. Here, ds is the number of discovered web services and ps is the number of preferred web services to be search by proposed algorithms. The precision and recall are given as Equation (1) and (2) respectively. Equation (3) describes the F-measure in general where β value is varied from 0.5 to 2. Rather than using only balanced F1-measure, we have also used F2-measure and F0.5-measure. F2 and F0.5 measures are the addition scores which returns the weighted recall and the weighted precision respectively. The F2-score shows results for increased weightage of discovered relevant web services while the F0.5-score describe results for increased relevancy of discovered web services during execution process. Table V and Table VI present the performance of service discovery with proposed strategy for NLR using OWL-S and WSMO dataset respectively.

$$P = \frac{ds \cap ps}{ds} \quad (1)$$

$$R = \frac{ds \cap ps}{ps} \quad (2)$$

$$F\beta = \frac{(1+\beta^2) * pd * rd}{(\beta^2 * pd) + rd} \quad (3)$$

The average execution time for service discovery with NLR using OWL-S and WSMO is 3.97 and 2.72 minutes respectively. Fig. 4 shows the graphs for (a) Precision-Recall, and (b) F1, F2 and F0.5 measure. In Fig. 4 and 5, P, R and Tc are the shorting of Precision, Recall and Test case respectively.

The results given in 4(a) describe that the average precision is better in almost all test cases than average recall value. It can be clearly seen that in Fig. 4(b), the F-0.5 measure has performed better than the F1-score and F2-score which means that the performance of relevant discovered services during execution process is better than the discovered relevant web services. It has also observed from 4(a) and 4(b) that the increasing size of dataset and NLRs have decreased the gap between average precision and average recall and F-measure. Further, Fig. 4(c) and 4(d) illustrates the average precision and average recall for positive and negative NLRs. The average precision and average recall values are higher in case of positive queries as compared to negative queries. The increasing connectives in negation-oriented NLRs have a deep impact on the average recall value for service discovery.

Next, in Fig. 5(a) and 5(b), the results corresponding to average precision, average recall and F-measure for WSMO test cases have been given. The average precision is higher in both the test cases (see Fig. 5(a)). In second test case, the value of average recall is less as compare to first test case while the number of web services as well as NLRS have increased in the second test case. This result can be justified with the fact that although the number of web services have increased in second test case but the increased amount of web services is just 25in number, which is influential for precision in terms of increased searched services but not for the relevant results i.e. recall value. In Fig. 5(b), the performance of F0.5-score is higher than the F1-score and F2-score. This result has same implications as for Fig. 4(b). The average precision and average

Table-V: Performance Measurement of Proposed Strategy with OWL-S TC V3 Dataset

Web Services		NLR (in %)		Performance Measurement (in average)				
Ratio	No. of Services	Ratio	No. of NLRs	Precision	Recall	F1	F2	F 0.5
30%	Education(77), Communication(17), Medical(22), Food(10), Travel(49), Geography(18), Economy(108)	30%	S +ve (9)	0.961	0.912	0.9110	0.9214	0.9508
			S -ve (12)	0.932	0.879	0.9047	0.8891	0.9209
			C^2 +ve (9)	0.942	0.869	0.9040	0.8827	0.9264
			C^2 -ve (12)	0.890	0.800	0.8426	0.8165	0.8704
			C^3+ve (10)	0.887	0.765	0.8215	0.7866	0.8596
			C^3 -ve (13)	0.857	0.731	0.7890	0.7531	0.8284
50%	Education(129), Communication(29), Medical(36), Food(17), Travel(82), Geography(30), Economy(180)	50%	S +ve (15)	0.924	0.909	0.9164	0.9120	0.9210
			S -ve (20)	0.901	0.828	0.8630	0.8416	0.8854
			C^2 +ve (15)	0.892	0.820	0.8545	0.8335	0.8766
			C^2 -ve (20)	0.840	0.787	0.8126	0.7971	0.8288
			C^3+ve (17)	0.835	0.759	0.7952	0.7731	0.8186
			C^3 -ve (22)	0.753	0.725	0.7387	0.7304	0.7472
70%	Education(181), Communication(41), Medical(51), Food(24),Travel(115), Geography(42), Economy(251)	70%	S +ve (21)	0.936	0.899	0.9171	0.9062	0.9284
			S -ve (28)	0.853	0.811	0.8315	0.8191	0.8443
			C^2 +ve (21)	0.903	0.818	0.8584	0.8337	0.8846
			C^2 -ve (28)	0.802	0.750	0.7751	0.7599	0.7910
			C^3+ve (24)	0.888	0.739	0.8067	0.7647	0.8536
			C^3 -ve (31)	0.719	0.711	0.7150	0.7126	0.7174
100%	Education(258), Communication(58), Medical(73), Food(34),Travel(165), Geography(60), Economy(359)	100%	S +ve (30)	0.936	0.921	0.9284	0.9240	0.9330
			S-ve (40)	0.832	0.809	0.8203	0.8135	0.8273
			C^2 +ve (30)	0.926	0.917	0.9215	0.9188	0.9242
			C^2 -ve (40)	0.789	0.753	0.7706	0.7599	0.7815
			C^3+ve (35)	0.899	0.872	0.8853	0.8773	0.8935
			C^3 -ve (45)	0.721	0.639	0.6775	0.6539	0.7030

recall values are higher in case of positive NLRs as compared to negative NLRs but the gap between average precision and average recall seems higher for WSMO test cases when compared with OWL-S test cases. From results, it can be said that the proposed approach is working well for S +ve , S –ve, C^2 +ve and C^2 –ve but have less average recall for C^3 +ve n C^3 –ve.

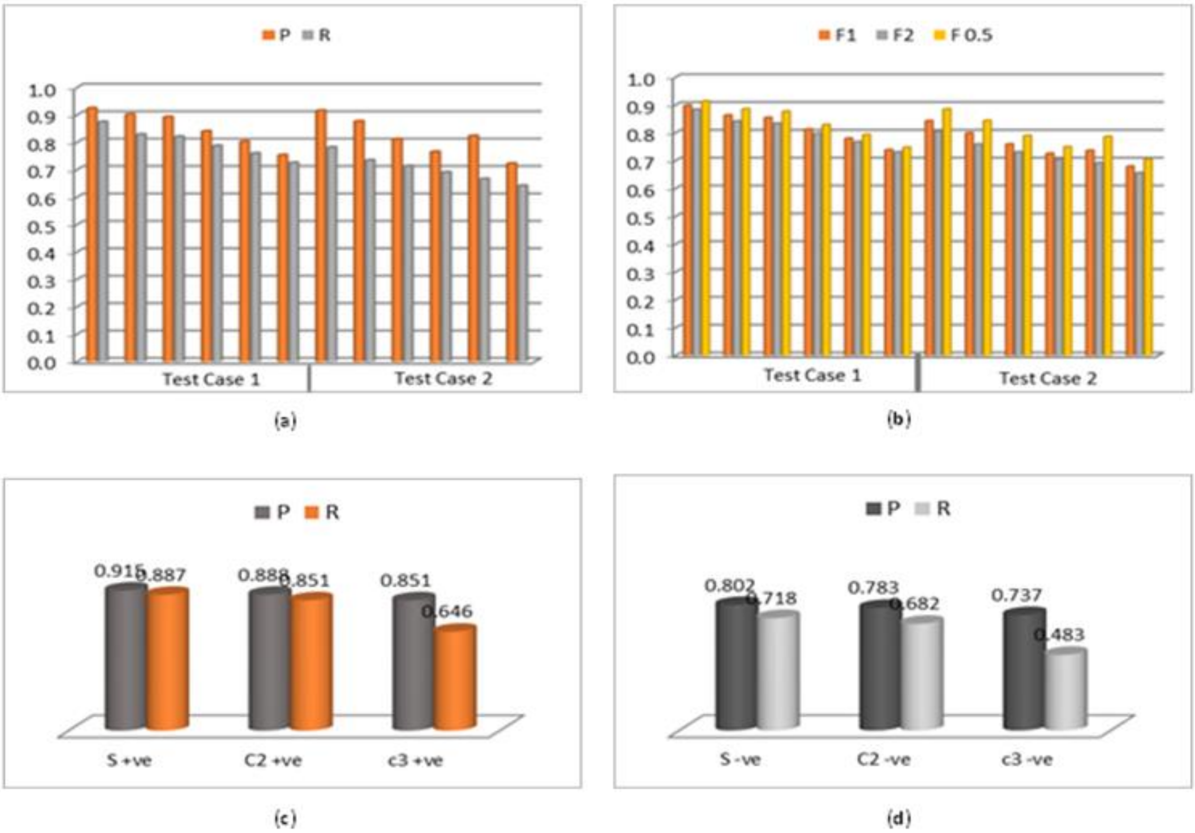


Fig. 4. Service discovery using OWL-S services: (a) Average Precision-Average Recall, (b) F1, F2 and F-0.5, (c) Average Precision-Average Recall for positive NLRs, (d) Average Precision-Average Recall

Table-VI: Performance Measurement of Proposed Strategy with WSMO Dataset

Web Services (No. of Services)	NLR (in %)		Performance Measurement (in average)				
	Ratio	No. of NLRs	Precision	Recall	F1	F2	F 0.5
Account Info(2), Bag On Net(3), Bin Validation(1), Breakeven point (1), Currency services (4), FedWire(3), Google Search(3), Prepaid credit card(5), Search service(2), Sports Good Finder(1), Store Manager(2), Survey(1), XWeb Blog(7), Cart Service (5), postal code& geography(5), Food Finder(5)	50%	S +ve (7)	0.924	0.874	0.8983	0.8836	0.9135
		S -ve (10)	0.901	0.828	0.8630	0.8416	0.8854
		C^2 +ve (9)	0.892	0.820	0.8545	0.8335	0.8766
		C^2 -ve (10)	0.840	0.787	0.8126	0.7971	0.8288
		C^3+ve (11)	0.802	0.759	0.7799	0.7672	0.7930
		C^3 -ve (12)	0.753	0.725	0.7387	0.7304	0.7472
	100%	S +ve (14)	0.915	0.781	0.8427	0.8046	0.8846
		S -ve (20)	0.876	0.734	0.7987	0.7586	0.8434
		C^2 +ve (18)	0.811	0.713	0.7588	0.7307	0.7893
		C^2 -ve (20)	0.765	0.690	0.7256	0.7038	0.7487
		C^3+ve (22)	0.823	0.666	0.7362	0.6924	0.7859
		C^3 -ve (25)	0.722	0.641	0.6791	0.6557	0.7042

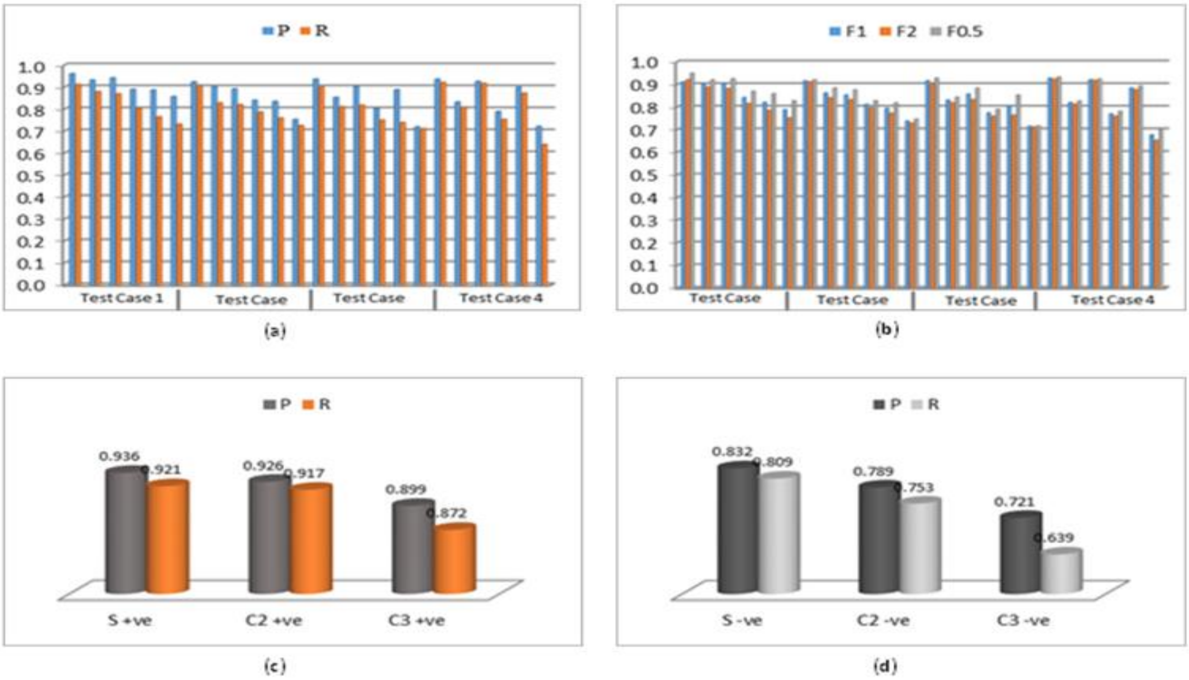


Fig. 5. Service discovery using WSMO services: (a) Average Precision-Average Recall, (b) F1, F2 and F-0.5, (c) Average Precision-Average Recall for positive NLRs, (d) Precision-Recall for negative NLRs.

Table- VII: Comparison with existing approaches (Sim.= Simple, Conj.= Conjunction, Cond.= Conditional, Neg.=Negation)

Method	NLR				Semantic Formalism		No. of NLRs
	Sim.	Conj.	Cond.	Neg.	OWL-S	WSMO	
Lim and Lee (2010)	✓	✓	✓	×	×	×	127
Adala et al. (2011)	----	×	×	×	----	×	---
Paulraj and Swamynathan. (2012)	✓	×	×	×	✓	×	42
Sanger et al. (2013)	✓	×	×	×	×	✓	61
Kamath & Ananthanarayana (2017)	✓	×	×	×	✓	×	240
Proposed Work	✓	✓	✓	✓	✓	✓	339

A comparison of proposed

approach with other existing approaches has been given in Table VII. The comparison is based upon three attributes i.e. types of NLR, the adopted semantic web service formalism and the total number of NLRs used for experimentation. In Table VII, we can clearly analyze the performance of proposed strategy in terms of coverage of formalism and complex queries.

VI. CONCLUSION

The right interpretation of NLR is a hammering problem in computational linguistics but still the popularity and demand for natural language processing has encouraged many researchers to propose their theoretical and practical solutions for this trend. This paper also proposes a solution for handling NLR for discovering the suitable single or composite semantic web services. The three contributions have been made in the current research article. The first contribution is to describe the complexity of NLR using simple, conjunctional, conditional connectives with positive or negative orientation in the form of Six-Slab range. The connectives have been used with positive and negative orientation of user's request. The next contribution is to put forward two algorithms for handling NLR, inclusion of semantic in NLRs and semantic matchmaking of NLR and web services. Further, the proposed strategy is implemented on two semantic formalisms i.e. OWL-S and WSMO. Four test cases for OWL-S and two test cases for WSMO web services have been executed. During implementation phase, the proposed strategy has achieved high precision and recall values for different NLRs. The overall outcome has showed that the proposed research work is performing better for simple, conditional and conjunctional service requests whereas the proposed strategy has to work in future for having better results in case of NLRs with more connectives i.e. C^3 +ve and C^3-ve.

In near future, the authors are committed to implement the proposed strategy for Generic Discovery Mechanism (GDM). The authors are also looking forward for handling the trade-off between complexity and efficient results (in similarity terms) as well as complexity and time. Next, the authors are also planning to work on service request subnet using psychological cognitive modeling because human cognition is robust to noise and can interpret sentences by dramatically changing the granularity level. As nouns, verbs and adjectives can easily be transformed to graphical network, so, in our opinion, experimentation with cognitive psychology concepts can improve service discovery.

ACKNOWLEDGMENT

The Authors would like to express their gratitude to the University Grant Commission (UGC), Govt. of India, to give financial support as Fellowship under UGC-NET for Junior Research Fellowship (JRF) scheme, to do this research effort. The Authors are also thankful to Department of Computer Science, Guru Nanak Dev University for providing infrastructure and other facilities for doing research work.

REFERENCES

1. G. O. Young, "Synthetic structure of industrial plastics (Book style with paper title and editor)," in *Plastics*, 2nd ed. vol. 3, J. Peters, Ed. New York: McGraw-Hill, 1964, pp. 15–64.
2. G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Jang, M., Sohn, J. C., Cho, H. K. (2007). Automated question answering using semantic web services. In Asia-Pacific Service Computing Conference, The 2nd IEEE(pp. 344-348). IEEE.
3. Heflin, J. (2004). OWL Web Ontology Language use cases and requirements. http://www. w3. org/TR/webont-req/.
4. Klusch, M., Kapahnke, P., Schulte, S., Lecue, F., Bernstein, A. (2016). Semantic web service search: A brief survey. KI-Künstliche Intelligenz, 30(2), 139-147.
5. García-Sánchez, F., Fernández-Breis, J. T., Valencia-García, R., Gómez, J. M., & Martínez-Béjar, R. (2008). Combining Semantic Web technologies with Multi-Agent Systems for integrated access to biological resources. Journal of Biomedical Informatics, 41(5), 848-859.
6. Farrag, T. A., Saleh, A. I., Ali, H. A. (2013). Toward SWS Discovery: Mapping from WSDL to OWL-S Based on Ontology Search and Standardization. IEEE Transactions on Knowledge and Data Engineering , 25 (5), 1135-1147
7. Lu, W., Cai, Y., Che, X., Lu, Y. (2016). Joint semantic similarity assessment with raw corpus and structured ontology for semantic-oriented service discovery. Personal and Ubiquitous Computing, 20(3), 311-323.
8. Narock, T., Yoon, V., March, S. (2014). A provenance-based approach to semantic web service description and discovery. Decision Support Systems, 64, 90-99.
9. Rodriguez-Mier, P., Pedrinaci, C., Lama, M., Mucientes, M. (2016). An integrated semantic web service discovery and composition framework. IEEE transactions on services computing, 9(4), 537-550.
10. Wang, W., De, S., Cassar, G., Moessner, K. (2015). An experimental study on geospatial indexing for sensor service discovery. Expert Systems with Applications, 42(7), 3528-3538.
11. Zhang, N., Wang, J., Ma, Y., He, K., Li, Z., Liu, X. F. (2018). Web service discovery based on goal-oriented query expansion. Journal of Systems and Software, 142, 73-91
12. Kamath, S. S., & Ananthanarayana, V. S. (2016). Semantics-based Web service classification using morphological analysis and ensemble learning techniques. International Journal of Data Science and Analytics, 2(1-2), 61-74.
13. Elbedweihy, K., Wrigley, S. N., Ciravegna, F. (2012a). Evaluating semantic search query approaches with expert and casual users. In International Semantic Web Conference (pp. 274-286). Springer, Berlin, Heidelberg.
14. Sangers, J., Frasincar, F., Hogenboom, F., Chepegin, V. (2013). Semantic web service discovery using natural language processing techniques. Expert Systems with Applications, 40(11), 4660-4671.
15. Adala, A., Tabbane, N., Tabbane, S. (2011). A framework for automatic web service discovery based on semantics and NLP techniques. Advances in Multimedia, 2011, 1.
16. Lim, J., Lee, K. H. (2010). Constructing composite web services from natural language requests. Web Semantics: Science, Services and Agents on the World Wide Web, 8(1), 1-13.
17. Katakis, I., Meditskos, G., Tsoumakas, G.,Bassiliades, N. (2009). On the combination of textual and semantic descriptions for automated semantic web service classification. In IFIP International Conference on Artificial Intelligence Applications and Innovations (pp. 95-104). Springer, Boston, MA.
18. Paredes-Valverde, M. A., Noguera-Arnaldos, J. Á., Rodríguez-Enríquez, C. A., Valencia-García, R., Alor-Hernández, G. (2015). A Natural Language Interface to Ontology-Based Knowledge Bases. In Distributed Computing and Artificial Intelligence, 12th International Conference (pp. 3-10). Springer, Cham.
19. Choueka, Y. (2014). Natural Language Processing Challenges, Achievements and Problems. In Language, Culture, Computation. Computational Linguistics and Linguistics (pp. 1-13). Springer, Berlin, Heidelberg.
20. Habernal, I., & KonopíK, M. (2013). SWSNL: semantic web search using natural language. Expert Systems with Applications, 40(9), 3649-3664.
21. Noessner, J., Niepert, M., Meilicke, C., Stuckenschmidt, H. (2010). Leveraging terminological structure for object reconciliation. In Extended Semantic Web Conference (pp. 334-348). Springer, Berlin, Heidelberg.

22. Demidova, E., Fankhauser, P., Zhou, X., Nejdl, W. (2010). DivQ: diversification for keyword search over structured databases. In Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval (pp. 331-338). ACM.
23. Klusch, M., Kapahnke, P., Fries, B., Khalid, M. A., Vasileski, M. (2018). OWLS-TC 3.0 – Release, Accessed on: 2 January 2018. Available at < http://projects.semwebcentral.org/projects/owls-tc/>.
24. Hopper, P. (1999). Short Course in Grammar. W. W. Norton Company.
25. Narayanan, R., Liu, B., Choudhary, A. (2009). Sentiment analysis of conditional sentences. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1 (pp. 180-189). Association for Computational Linguistics.
26. Iqbal, R., Murad, M. A. A., Selamat, M. H., & Azman, A. (2012). Negation query handling engine for natural language interfaces to ontologies. In Information Retrieval & Knowledge Management (CAMP), 2012 International Conference on (pp. 249-253). IEEE.
27. Iqbal, R., & Murad, M. A. A. (2013). A Negation Query Engine for Complex Query Transformations. Journal of Science and Technology, 193-204.
28. Jordan, M. P. (1998). The power of negation in English: Text, context and relevance. Journal of Pragmatics, 29(6), 705-752.
29. Bosca, A., Corno, F., Valetto, G., Maglione, R. (2006). On-the-fly construction of web services compositions from natural language requests. JSW, 1(1), 40-50.
30. Al-Muhammed, M. J., Embley, D. W. (2006). Resolving underconstrained and overconstrained systems of conjunctive constraints for service requests. In International Conference on Advanced Information Systems Engineering(pp. 223-238). Springer, Berlin, Heidelberg.
31. Xie, F., Gong, H., Deng, D., Wang, S., Wang, G. T., Hu, J., Sheu, P. C. (2006). Integrating semantic web services for declarative accesses in natural language. In Multimedia, 2006. ISM'06. Eighth IEEE International Symposium on (pp. 201-208). IEEE.
32. Englmeier, K., Pereira, J.., Mothe, J. (2006). Choreography of web services based on natural language storybooks. In Proceedings of the 8th international conference on Electronic commerce: The new e-commerce: innovations for conquering current barriers, obstacles and limitations to conducting successful business on the internet (pp. 132-138). ACM.
33. Tablan, V., Damljanovic, D., Bontcheva, K. (2008). A natural language query interface to structured information. In European Semantic Web Conference (pp. 361-375). Springer, Berlin, Heidelberg.
34. Booth, J., Di Eugenio, B., Cruz, I. F., Wolfson, O. (2009). Query sentences as semantic (sub) networks. In Semantic Computing, 2009. ICSC'09. IEEE International Conference on (pp. 89-94). IEEE.
35. Pop, F. C., Cremene, M., Vaida, M., & Riveill, M. (2010a). Natural language service composition with request disambiguation. In International Conference on Service-Oriented Computing (pp. 670-677). Springer, Berlin, Heidelberg.
36. Pop, F. C., Cremene, M., Vaida, M., Riveill, M., Tigli, J. Y., & Lavirotte, S. (2010b). Natural language based on-demand service composition. International Journal of Computers Communications & Control, 5(5), 871-883.
37. Cremene, M., Tigli, J. Y., Lavirotte, S., Pop, F. C., Riveill, M., Rey, G. (2009). Service composition based on natural language requests. In Services Computing, 2009. SCC'09. IEEE International Conference on (pp. 486-489). IEEE.
38. Paulraj, D., Swamynathan, S. (2011). Content based service discovery in semantic web services using wordnet. In International Conference on Advanced Computing, Networking and Security (pp. 48-56). Springer, Berlin, Heidelberg.
39. Roy, S. D., Zeng, W. (2013). Cognitive canonicalization of natural language queries using semantic strata. ACM Transactions on Speech and Language Processing (TSLP), 10(4), 20.
40. Kamath, S., & Ananthanarayana, V. S. Discovering composable web services using functional semantics and service dependencies based on natural language requests. Information Systems Frontiers, 1-15.
41. Miller, G. A. (1995). WordNet: A Lexical Database for English. Communications of the ACM , 39-41.
42. Walenz, B., Didion, J. (2011). JWNL: Java WordNet library. Available from <http://sourceforge.net/projects/jwordnet/>.
43. Finlayson, M. (2012). JWI: The MIT java WordNet interface. Available from <http://projects.csail.mit.edu/jwi/>.
44. Negi A., Kaur P. (2019) Examination of Sense Significance in Semantic Web Services Discovery. In: Krishna A., Srikantaiah K., Naveena C. (eds) Integrated Intelligent Computing, Communication and Security. Studies in Computational Intelligence, vol 771. Springer, Singapore
45. Banerjee S, Pedersen T (2003) Extended gloss overlaps as a measure of semantic relatedness. In: 18th joint conference on Artificial intelligence. Morgan Kaufmann Publishers, San Francisco, pp. 805-810.
46. Chen F, Lu C, Wu H, Li M (2017) A semantic similarity measure integrating multiple conceptual relationships for web service discovery. Expert Systems with Applications 67:19-31. doi: 10.1016/j.eswa.2016.09.028

AUTHORS PROFILE

Aradhana Negi is a Ph.D. research scholar in the Department of Computer Science, Guru Nanak Dev University, India. She received her Master of Computer Applications from Guru Nanak Dev Engineering College, India. Her research interest includes Semantic Web, Automatic Semantic Web Service Discovery and Composition, and Cognitive Psychology in computing.

Parminder Kaur is working as Assistant Professor in the Department of Computer Science, Guru Nanak Dev University, India. She has received her Ph.D. degree in 2011 from Department of Computer Science, Guru Nanak Dev University, India. Her research interests are Component-Based Software Engineering, Semantic Web Services, Version Controlling mechanism in Open Source Development System. She has published approximately 46 research papers in various international/national journals and conferences.