AN IMPROVED ACCURACY OF WEB SERVICE SELECTION BASED ON MULTI-CRITERIA DECISION MAKING AND WEB SERVICE MODELING ONTOLOGY

MOJTABA KHEZRIAN

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Doctor of Philosophy (Computer Science)

Faculty of Computing
Universiti Teknologi Malaysia

JULY 2013

I declare that this thesis entitled "*An Improved Accuracy of Web Service Selection Based on Multi-Criteria Decision Making and Web Service Modelling Ontology"* is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : ……………………….

Name : Mojtaba Khezrian

Date : 26 JULY 2013

To Mahdi, the promised saviour,

looking forward to his arrival….

ACKNOWLEDGEMENT

First, I would like to thank God, the most gracious and the most merciful. Praise be to God who created us and gave us intelligence and guidance. Peace be upon our prophet, the teacher of all mankind.

I would like to thank and convey my sincere appreciation to my principal supervisor, Assoc. Prof. Dr Wan Mohd Nasir Wan Kadir, for his encouragement, guidance and support. I would also like to thank my co-supervisor, Assoc. Prof. Dr Suhaimi Ibrahim, who supported me in all levels of my study. Moreover, I owe so much to Dr. Sayed Gholam Hassan Tabatabaei for his support and kind.

I would also like to thank the Universiti Teknologi Malaysia (UTM) for providing me with all the requirements and needs of a research student. I would like to give special thanks and appreciation to the Ministry of Higher Education (MOHE) for funding my Ph.D. studies.

I would particularly like to thank my parents and my wife's parents, who deserve my gratitude for their inseparable prayer, encouragement and endless patience. Words fail me in expressing my deepest appreciation to my wife, whose dedication, love and support gave me confidence. My thesis would not have been possible without her patience and encouragement. Thank you.

ABSTRACT

The subject of web services has become a popular topic in the area of computer science, as it provides the ability to collect capabilities and components in a unique interface to meet user requirements. One of the significant issues in this area is the development of an accurate service selection approach. In the existing approaches, accuracy refers to the accuracy of the selection method; the accuracy of the input data are neglected. There are many approaches in service selection for managing modelling or algorithmic issues. This research proposes an Accurate approach based on multi-criteria decision making (MCDM) and web service modelling ontology (WSMO), which is called AMW. The accuracy of the selection method is improved compared to existing methods, and the accuracy of the input data are considered. For this purpose, first, a comparative evaluation of state-of-the-art approaches for web service selection approaches has been performed, and the strengths and weaknesses of those approaches have been discussed. Second, the weaknesses of the existing approaches have been identified by applying the analytical hierarchy process (AHP) method to define default criteria weights and enhanced VIšekriterijumsko KOmpromisno Rangiranje (VIKOR) for the selection of services. Moreover, to improve the accuracy of input data, the confidence level of the service provider and the power of the decision maker are considered. Finally, the AMW approach has been validated by applying two case studies with various situations. The results of the experimental validation demonstrate that AMW provides an accurate and feasible solution. The results of this research can assist service consumers in attaining a more accurate decision when selecting the appropriate service.

ABSTRAK

Perkhidmatan web telah menjadi satu topik yang popular dalam bidang sains komputer, kerana ia menyediakan kebolehan untuk mengumpul keupayaan dan komponen dalam antara muka yang unik untuk memenuhi keperluan pengguna. Salah satu isu penting dalam bidang ini adalah pembangunan pendekatan pemilihan perkhidmatan yang tepat. Dalam pendekatan yang sedia ada, ketepatan merujuk kepada ketepatan kaedah pemilihan; ketepatan data input adalah diabaikan. Terdapat banyak pendekatan dalam pemilihan perkhidmatan untuk menguruskan isu-isu model atau algoritma. Kajian ini mencadangkan satu pendekatan yang tepat berdasarkan penghasilan keputusan multi-kriteria (MCDM) dan permodelan ontologi perkhidmatan web (WSMO) yang dipanggil AMW. Ketepatan kaedah pemilihan adalah lebih baik berbanding dengan kaedah yang sedia ada, dan ketepatan data input telah dipertimbangkan. Bagi tujuan ini, penilaian perbandingan terhadap pendekatan terkini dalam pemilihan perkhidmatan web telah dilaksanakan, dan kekuatan dan kelemahan pendekatan tersebut telah dibincangkan. Kemudian, kelemahan pendekatan sedia ada telah dikenal pasti dengan menggunakan kaedah proses hierarki analisis (AHP) untuk menentukan pemberat kriteria lalai dan VIšekriterijumsko KOmpromisno Rangiranje (VIKOR) yang dipertingkatkan untuk pemilihan perkhidmatan. Selain itu, untuk meningkatkan ketepatan data input, tahap keyakinan pembekal perkhidmatan dan kuasa pembuat keputusan akan dipertimbangkan. Akhirnya, pendekatan AMW telah disahkan dengan melaksanakan AMW terhadap dua kajian kes dengan pelbagai situasi berbeza. Keputusan pengesahan melalui eksperimen menunjukkan bahawa AMW menyediakan penyelesaian yang tepat dan boleh dilaksanakan. Hasil kajian ini boleh membantu pengguna perkhidmatan dalam mencapai keputusan yang lebih tepat dalam memilih perkhidmatan yang sesuai.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE
	ACKNOWLEDGEMENT	iv
	ABSTRACT	v
	ABSTRAK	vi
	TABLE OF CONTENTS	vii
	TABLE OF TABLES	xi
	TABLE OF FIGURES	xiv
	LIST OF ABBREVIATIONS	xvii
	LIST OF APPENDICES	xviii
1	**INTRODUCTION**	1
	1.1 Background of the Problem	1
	1.2 Statement of the Problem	2
	1.3 Objectives of the Study	4
	1.4 Scope of the Study	4
	1.5 Significance of the Study	5
	1.6 Thesis Organization	6
2	**LITERATURE REVIEW**	8
	2.1 Web Services	8
	2.1.1 SOAP	9
	2.1.2 WSDL	10
	2.1.3 UDDI	11
	2.2 Semantic	12
	2.2.1 Ontology	12

		2.2.2 Semantic Web	13
		2.2.3 Semantic Web Services	16
	2.3	MCDM	16
		2.3.1 AHP	17
		2.3.2 TOPSIS	19
		2.3.3 VIKOR	21
	2.4	Web Service Selection (WSS)	25
		2.4.1 General Framework for WSS	27
		2.4.2 QoS for WSS	28
	2.5	Approaches in WSS	30
		2.5.1 Policy-based Approach	31
		2.5.2 Trust and Reputation-based Approach	32
		2.5.3 Improve Protocol-based Approach	33
		2.5.4 Semantic-based Approach	34
		2.5.5 Decision Making	43
	2.6	Summary	46
3	**COMPARATIVE EVALUATION OF WSS APPROACHES**		47
	3.1	WSS Criteria	47
		3.1.1 User Preference	47
		3.1.2 Performance	48
		3.1.3 Accuracy	48
		3.1.4 Automation	48
		3.1.5 Scalability	49
	3.2	Comparison at the First Level	49
	3.3	Comparison at the Second Level	52
		3.3.1 Semantic Level	52
		3.3.2 MCDM Level	53
	3.4	Summary	55
4	**RESEARCH METHODOLOGY**		56
	4.1	Research Design	56

	4.2	Research Procedure	57
		4.2.1 Literature Review	57
		4.2.2 Analysis of the Requirements	59
		4.2.3 Development	60
		4.2.4 Evaluation	60
	4.3	Operational Framework	67
	4.4	Instrumentation	67
	4.5	Assumptions and Limitations	69
	4.6	Summary	69
5	**THE AMW APPROACH TO SUPPORT WSS**		70
	5.1	Limitations and Restrictions	70
	5.2	The AMW Approach	72
	5.3	The Architectural Aspect of AMW	73
		5.3.1 AMW Components	75
		5.3.2 Abstract View of the Service Selection Mechanism	78
		5.3.3 Concrete View of the WSS Mechanism	80
	5.4	The Algorithmic Aspect of AMW	84
		5.4.1 Pre-selection	84
		5.4.2 Selection	90
		5.4.3 Post-selection	93
	5.5	Goal Generator	96
		5.5.1 Goal WSML File	96
		5.5.2 Flowchart of the Goal Generator	97
		5.5.3 Algorithm of the Goal Generator	99
	5.6	AMW Prototype Implementation	100
		5.6.1 Pre-selection	100
		5.6.2 Selection	103
		5.6.3 Post-selection	105
	5.7	Summary	106

6	**EVALUATION OF THE PROPOSED APPROACH**	108
	6.1 Case Study	108
	6.1.1 Flight Booking	110
	6.1.2 Hotel Reservation	124
	6.2 Results of the Existing Approach	138
	6.2.1 Flight Booking	139
	6.2.2 Hotel Reservation	139
	6.3 Analysis Result of Case Studies	140
	6.4 Experimental Result	142
	6.5 Qualitative Analysis Result	150
	6.5.1 Feature Identification	150
	6.5.2 Feature Scoring	151
	6.5.3 Analysis	154
	6.6 Summary	157
7	**CONCLUSION**	159
	7.1 Summary and Achievement	159
	7.2 Contributions of the Research	162
	7.3 Future Work	162
REFERENCES		164
APPENDIX A		171

TABLE OF TABLES

TABLE NO.	TITLE	PAGE
2.1	The Saaty rating scale [28]	18
3.1	Comparative evaluation of WSS at the first level	50
3.2	Comparison at the semantic level	52
3.3	Comparison at the MCDM level	54
3.4	Summary of the MCDM approaches	54
4.1	Operational Framework	68
5.1	Mapping the goal generator table	76
6.1	Possible states of the AMW approach	109
6.2	Achievable scenarios and assigned case study	109
6.3	Rates of the alternatives from the SP	112
6.4	Confidence level of service providers	113
6.5	Rates of the alternatives from the SP with a CL	113
6.6	Rates of alternatives from the expert DMs	113
6.7	Power of DMs	114
6.8	Rates of the alternatives from the expert DMs with power of the DMs	114
6.9	Aggregated rates of the alternatives	114
6.10	Final rates of the alternatives with *TOM* and *RS*	115
6.11	Best and worst values for all criteria	115
6.12	Normalised decision matrix	116

6.13	Values of Si and Ri	116
6.14	Ranking of the alternatives	117
6.15	Calculated similarity rates	117
6.16	Estimated CL	118
6.17	Final rates of the alternatives with TOM and RS	119
6.18	Values of Si and Ri	119
6.19	Ranking of the alternatives	120
6.20	Aggregated rates without the CL of the SP	121
6.21	Best and worst values for all criteria	122
6.22	Normalised decision matrix	122
6.23	Values of Si and Ri	122
6.24	Ranking of the alternatives	123
6.25	Rates of the alternatives from the SP	125
6.26	CLs of the SPs	125
6.27	Rates of the alternatives from SPs with CLs	125
6.28	Values of the service selection factors in the 11-point scale format	126
6.29	Linguistic rates of the alternatives from the expert DMs	126
6.30	Power of the expert DMs	127
6.31	Numeric aggregated rates of the alternatives from the expert DMs	127
6.32	Final average rates of the alternatives	127
6.33	Final rates of the alternatives with TOM and RS	128
6.34	Best and worst values for all criteria	128
6.35	Normalised decision matrix	129
6.36	Values of Si and Ri	129
6.37	Ranking of the alternatives	130

6.38	Calculated similarity rates	130
6.39	Estimated CLs	131
6.40	Rates of the alternatives from the SP	131
6.41	Final rates of alternatives with *TOM* and *RS*	132
6.42	Best and worst values for all criteria	132
6.43	Normalised decision matrix	132
6.44	Values of $\boldsymbol{Si}$ and $\boldsymbol{Ri}$	133
6.45	Ranking of the alternatives	133
6.46	Aggregated rates of alternatives with *TOM* and *RS*	135
6.47	Final rates of the alternatives with *TOM* and *RS*	135
6.48	Best and worst values for all criteria	136
6.49	Normalised decision matrix	136
6.50	Values of $\boldsymbol{Si}$ and $\boldsymbol{Ri}$	136
6.51	Ranking of the alternatives	137
6.52	Non-functional values of the alternatives	139
6.53	Non-functional values of the alternatives	140
6.54	Importance of features	152
6.55	Assessment table for AMW and WSMO	154
6.56	Average evaluation profile for AMW and WSMO	155

TABLE OF FIGURES

FIGURE NO.	TITLE	PAGE
2.1	Web Service Architecture	9
2.2	SOAP message	10
2.3	Example of a WSDL file	11
2.4	Simple ontology	13
2.5	Simple RDF model	14
2.6	RDF example in XML format	15
2.7	Excerpt of a simple ontology written in OWL	15
2.8	Matrix for evaluating criteria weights	18
2.9	Ideal and compromise solutions	22
2.10	Discovery, selection and composition of services	26
2.11	General framework of WSS	27
2.12	WSS classification	31
2.13	Core WSMO elements	34
2.14	Type of WSML	35
2.15	OWL-S Conceptual Model	38
2.16	Architecture of the semantic matcher	39
2.17	Interface components of SAWSDL	41
2.18	Matching level interface of SAWSDL-MX	42
4.1	Research Design	58

4.2	Direction of research based on the evaluation	59
4.3	General concepts of precision and recall	62
4.4	precision-recall graph	62
4.5	Sample of averaged 11-point precision-recall graph	63
4.6	Research Procedure	66
5.1	General architecture of the AMW approach	73
5.2	Flow chart of the proposed framework	79
5.3	Flow chart of the pre-selection stage	81
5.4	Flow chart of the selection stage based on the enhanced VIKOR algorithm	82
5.5	Flow chart of the post-selection stage	83
5.6	Defined criteria weights	86
5.7	Pair-wise comparing matrix	87
5.8	Defined rates of alternatives	89
5.9	WSS based on enhanced VIKOR	93
5.10	The Algorithm of estimating CL of SP	94
5.11	The algorithm of estimating the reputation of a service.	95
5.12	A sample of goal in WSML format.	96
5.13	Flowchart of the goal generator.	98
5.14	Goal generator algorithm.	99
5.15	The main GUI of the AMW prototype.	101
5.16	The goal generator used to translate user preferences to WSML format.	102
5.17	Extract and integrate accurate data and add it to the repository.	104
5.18	Screenshot of the AMW service selection.	105
5.19	Estimated reputation of the service after AMW service selection	106
6.1	Relationship between the criteria and services.	111

6.2	Averaged 11-point precision-recall graph across 6 scenarios applied	141
6.3	The 11-point interpolated precision-recall graph for query 1	144
6.4	The 11-point interpolated precision-recall graph for query 2	145
6.5	The 11-point interpolated precision-recall graph for query 3	146
6.6	The 11-point interpolated precision-recall graph for query 4	147
6.7	The 11-point interpolated precision-recall graph for query 5	148
6.8	The 11-point interpolated precision-recall graph for query 6	149
6.9	Averaged 11-point interpolated precision-recall graph across all queries	149
6.10	Graph of the importance of features	153
6.11	Assessment scale used to assess tool support for a feature	153
6.12	Final Evaluation profile	156
6.13	Final multiple-metric graph for AMW and WSMO	157

LIST OF ABBREVIATIONS

AMW	-	Accurate Approach based on MCDM and WSMO
CL	-	Confidence Level
DM	-	Decision Maker
MCDM	-	Multi Criteria Decision Making
OWL	-	Web Ontology Language
OWL-S	-	Web Ontology Language for Web Services
QoS	-	Quality of Service
SOA	-	Service Oriented Architecture
SOAP	-	Simple Object Access Protocol
SP	-	Service Provider
SWS	-	Semantic Web Service
UDDI	-	Universal Description, Discovery, and Integration
WSDL	-	Web Services Description Language
WSML	-	Web Service Modelling Language
WSMO	-	Web Service Modelling Ontology
WSMX	-	Web Service Modelling execution environment
WSS	-	Web Service Selection

LIST OF APPENDICES

APPENDIX	TITLE	PAGE
A	List of Publications	171

CHAPTER 1

INTRODUCTION

In this chapter, the introduction of research is discussed in detail. First, background of the problem is described. Then, to clarify the problem, the statement of the problem is provided. Moreover, the objectives and scope of the study are defined, and the significance of the study is discussed.

1.1 Background of the Problem

Researchers have recently shown increased interest in web services, which are among the most widely used groups in service-oriented architecture (SOA) and service computing. According to the World Wide Web Consortium (W3C), “A web service is a software system designed to support interoperable machine-to-machine interaction over a network” [1].

Many organisations and companies develop applications that are accessible via the Internet. Therefore, the capability of correctly selecting and combining inter-organisational and various services at runtime on the web is a significant issue in the development of web service applications [2].

The components of the traditional web service architecture are web service definition language (WSDL), simple object access protocol (SOAP) and universal description discovery and integration (UDDI), which are used to describe services, transfer messages and store services, respectively [3].

As described in recent research, web service mechanisms can be separated into discovery, selection and composition [4-7]. Web service discovery enables providers to publish service descriptions and profile information regarding businesses, services and other related details in UDDI repositories. However, there are instances in which non-functional properties need to be utilised and the most appropriate service needs to be selected to cater to user requirements, apart from functional properties. The selection component is used for this purpose. Finally, web service composition composes the selected services together within the time frame required. A set of services can be composed as a composite service to provide requisite functions [8]. This research concerns web service selection (WSS).

WSS appears when there is a set of discovered web services that can fulfil user requirements [9] and one of these services should be selected to be returned to the service consumer [10]. This selection must be tailored to user preferences because one user may require high quality, whereas another may require low prices [11].

At the present time, there are some approaches for sustaining semantic WSS, such as web service modelling ontology (WSMO) [4], web ontology language for web services (OWL-S) [12] and semantic annotations for WSDL (SAWSDL) [13]. The infrastructure of the proposed approach is WSMO, which is a suitable modelling ontology for supporting semantic web services in this research because it supports user preferences and can include non-functional properties in a straightforward manner.

1.2 Statement of the Problem

There are several approaches for WSS; these approaches use different methods for selecting the services that sustain the user requirements. The existing approaches attempt to improve the accuracy of WSS by improving the selection method. Although the method must be accurate, the accuracy of input data are more important, as accurate results cannot be obtained without accurate data. Existing approaches also attempt to propose a solution for the weighting of criteria. These approaches do not consider default criteria weights. However, default weights are very important when the service consumer is not familiar with the weighting system.

This research intends to provide an approach to select the most appropriate web service that fulfils the descriptions by WSMO. This research proposes a new service selection approach called AMW (Accurate approach based on multi-criteria decision making (MCDM) and WSMO) and posits that utilising AMW for service selection resolves the aforementioned issues. The hypothesis responds to the general research question:

How can MCDM and WSMO be used to achieve an accurate approach for selecting appropriate web services to respond to user requirements?

To answer this question, the following research questions must be addressed:

(i) Why are the existing approaches unable to answer the present problem in WSS?

(ii) How can accurate data be prepared, using extra QoS, and achieve the best results, particularly in terms of the accuracy of supporting WSS?

(iii) What are the main elements of the accurate approach for selecting web services?

(iv) How to select best service via the accurate approach?

(v) How to validate and evaluate the effectiveness of the proposed approach to support WSS?

1.3 Objectives of the Study

Based on the described problem statements, the research objectives are as follows.

(i) To investigate and evaluate the state of the art in web service selection approaches.

(ii) To develop a new approach for web service selection in which the accuracy of both the input data and selection method are considered.

(iii) To propose default criteria weights using the AHP method to help users express their preferences.

(iv) To design and formulate algorithms to support the proposed approach.

(v) To evaluate the accuracy of the proposed approach by developing a selector tool and comparing it with other approaches.

1.4 Scope of the Study

The three areas related to this subject are: semantic web services, MCDM and trust and reputation. These areas are described below.

First, this research is related to semantic web services. Web services are described semantically in semantic web services, and the discovery, composition and selection of services are completed via semantic web services. To describe a service, the capability and the required inputs, semantic-rich languages are used. For example, in making concept ontology service descriptions, the resource description framework (RDF) and OWL are used. The basis of the semantic web service of this research is WSMO. In Section 2.2.3, semantic web services are described in detail.

The second direction of this research is MCDM. The problem of service selection is similar to the MCDM problem. Therefore, one of the topics of this research is MCDM. There are some MCDM methods that can be applied in service selection problem, such as AHP, TOPSIS and VIKOR. This research employs VIKOR, which is absent in the service selection literature but is well known in other research areas. This subject is discussed in detail in Section 2.3.

Finally, trust and reputation are the other areas that are related to the proposed approach. Trust and reputation rely on feedback from other users who have used the service previously. In this situation, each service can be popular after it is used. In fact, it is the reputation of services that is gained by trusting service providers (SPs). Trust and reputation are described further in Section 2.5.2

1.5 Significance of the Study

Whereas some topics, such as discovery and composition for semantic web services, have been addressed by numerous studies, WSS as one of the final stages is essential. However, WSS has not been given sufficient attention; thus, additional research is necessary.

Although discovery and composition are important issues in web service studies, WSS is more important because if discovery and composition fulfil their tasks, one step still remains to fulfil the web service process; this step is the selection of the best discovered services that have functionalities similar to the user preferences. Therefore, the selection mechanism should perform this essential task. Consequently, successful discovery and composition stages would not yield a good result without an adequate selection mechanism.

WSS is one of the most significant discussions in SOA. WSS is the identification of the best candidate services among a group of services with similar functions but different Qualities of Service (QoS) [10]. QoS is important when quality metrics need to be accomplished through service stipulation. These metrics are measurable and include what service is being offered [14].

There are approaches available for supporting semantic WSS, such as WSMO [4], OWL-S [12, 15] and SAWSDL [13, 16]. However, these approaches attempt to improve the selection methods; the accuracy of data are not considered. In this research, the WSMO is used as the basis of the research to improve the accuracy of both the input data and selection method.

1.6 Thesis Organization

The organization of thesis is as follow: Literature Review (Chapter 2), Comparative Evaluation of WSS Approaches (Chapter 3), Research Methodology (Chapter 4), THE AMW Approach to Support WSS (Chapter 5), Evaluation of the Proposed Approach (Chapter 6) and Conclusion (Chapter 7).

In Chapter 2, the concepts of web services, semantic web and semantic web services are described. In addition, the WSS and related approaches are discussed. Moreover, MCDM and some related methods are described. Finally, the classification of WSS approaches is proposed, and each approach is described in detail.

Chapter 3 describes a comparative evaluation of the state-the-art approaches that are discussed in Chapter 2. First, the WSS criteria are described. Second, the comparison in the first level of classification is discussed. Third, the comparison of the second level, which is divided into the semantic level and MCDM level, is discussed. Finally, the approaches are evaluated.

Chapter 4 describes the methodology of this research, including the research design, research procedure, operational framework, instrumentation, assumptions and limitations.

In Chapter 5, the proposed approach, called AMW, is described. This approach fixes the issues discussed in Chapter 3. The proposed solution for providing flexible and automated service selection involves the application of a flexible framework and proficient MCDM method. The AMW approach involves a framework as an architectural aspect and a formula as an algorithmic aspect.

Chapter 6 describes the evaluation of the AMW approach. The aim of this section is to identify the types of problems that AMW can solve that cannot be solved by the existing approaches. The proposed approach must be validated analytically. Specifically, the conditions in which each of the existing approaches fails must be explained and specified, and the reasons for which AMW worked correctly also must be explained. The explanations are validated by demonstrations of each approach applied to specific examples that illustrate each of the conditions identified in the analysis. The results demonstrate how the features of AMW can affect the accuracy of WSS.

Chapter 7 presents the conclusions of this research. The achievements and contributions of the proposed approach, AMW, are summarised, and unresolved issues and future work are described.

CHAPTER 2

LITERATURE REVIEW

In this chapter, the concepts of web services, semantic web and semantic web services are described. In addition, the WSS and related approaches are discussed. Moreover, MCDM and some related methods are described. Finally, the classification of WSS approaches is proposed, and each approach is described in detail.

2.1 Web Services

The topic of web services has become a popular area of computer science. Web services allow the collection of capabilities and components in a unique interface to respond the user requirements. If functions and components cannot respond to the user, they can be merged to satisfy the user requirements. According to W3C, "a web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine processable format (specifically WSDL). Other systems interact with the web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialisation in conjunction with other web-related standards"[17].

Web services provide access functionality through the web using a collection of free principles that create the communication free of the operating system platform and programming language. Another definition of web service is a technology that allows applications to communicate with each other in a platform-independent manner [18].

The architecture of web services is presented in Figure 2.1 [19].

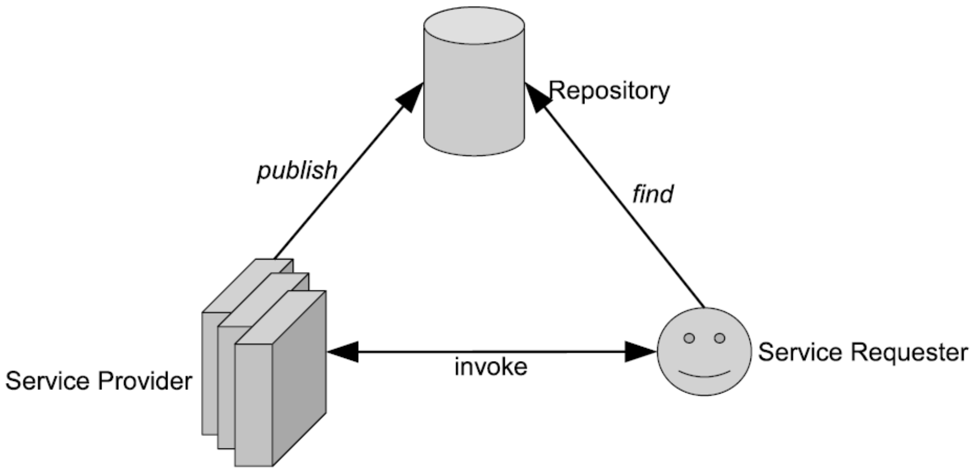

Figure 2.1 Web Service Architecture

Initially, to discover a service, a service provider must first bring out publish the service in the repository. After publishing the service, the service requester can send a query to obtain the service address. Additionally, the UDDI orchestrates this relation. To use the service, the requester must determine how to invoke the service that causes the WSDL to provide information[20]. Finally, to transfer data, the SOAP standard is used [19]. These standards are described below.

2.1.1 SOAP

Simple Object Access Protocol (SOAP) is a protocol that is used to transfer messages between applications via the web. According to W3C, "SOAP is a lightweight protocol intended for exchanging structured information in a decentralised, distributed environment. It uses XML technologies to define an extensible messaging framework providing a message construct that can be exchanged over a variety of underlying protocols. The framework has been designed to be independent of any particular programming model and other implementation specific semantics".[21]. A SOAP message is an XML document. A simple example of a SOAP message is provided in Figure 2.2 [21].

```
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">
 <env:Header>
  <n:alertcontrol xmlns:n="http://example.org/alertcontrol">
   <n:priority>1</n:priority>
   <n:expires>2001-06-22T14:00:00-05:00</n:expires>
  </n:alertcontrol>
 </env:Header>
 <env:Body>
  <m:alert xmlns:m="http://example.org/alert">
   <m:msg>Pick up Mary at school at 2pm</m:msg>
  </m:alert>
 </env:Body>
</env:Envelope>
```

Figure 2.2 SOAP message

2.1.2 WSDL

Web Service Definition Language (WSDL) is an XML document that is used to describe software component interfaces[19]. Fundamentally, the operations, inputs and output messages of services are described by WSDL. These operations and messages are expressed conceptually. WSDL [20] defines a service by six major elements: types, service, binding, port type, port and message. Additionally, WSDL allows for the division of the description of functionality offered by a service from existing details [22]. An example of a WSDL file is provided in Figure 2.3 [19].

```
[...]
<wsdl:types>
[...]
<s:element name=”GetWeather”>
<s:complexType>
<s:sequence>
<s:element minOccurs=”0” maxOccurs=”1” name=”CityName” type=”s:string” />
<s:element minOccurs=”0” maxOccurs=”1” name=”CountryName” type=”s:string” />
</s:sequence>
</s:complexType>
</s:element>
[...]
</wsdl:types>
<wsdl:message name=”GetWeatherIn”>
<wsdl:part name=”parameters” element=”tns:GetWeather” />
</wsdl:message>
[...]
<wsdl:portType name=”GlobalWeather”>
<wsdl:operation name=”GetWeather”>
<wsdl:input message=”tns:GetWeatherSoapIn” />
<wsdl:output message=”tns:GetWeatherSoapOut” />
</wsdl:operation>
[...]
```

Figure 2.3 Example of a WSDL file

2.1.3 UDDI

Descriptions of services and companies in XML format are collected by the Universal Description Discovery and Integration (UDDI) registry. UDDI is a business directory of web service interfaces stored in WSDL format. Other business information (e.g., contact information) can be stored in the UDDI registry. UDDI uses SOAP as its message exchange protocol.

According to [23], “UDDI is a mechanism for clients to dynamically find other web services”. Connecting businesses by external business partners to services provided is possible using a UDDI interface. The clients of a UDDI registry are divided into two types: (1) clients that want to find certain services to achieve their goal and bind to them programmatically and (2) businesses that publish a service. UDDI is layered over SOAP, and UDDI is assumed to include two types of objects, namely, responses and requests, which are sent as SOAP messages.

2.2 Semantic

In this section, semantic issues related to this research, namely, ontologies, semantic web and semantic web services, are discussed.

2.2.1 Ontology

“Ontology is a set of concepts, their properties and the relationships between them. Ontologies provide the building blocks for expressing semantics in a well-defined manner” [24].

To clarify the concept of ontologies, an example is shown in Figure 2.4. In this figure, ovals represent ontological concepts, and lines between the ovals represent relationships. In this example, the “subclass of” are used for relationships between concepts. Properties are represented by rectangles and are connected to concepts via dotted lines[25].

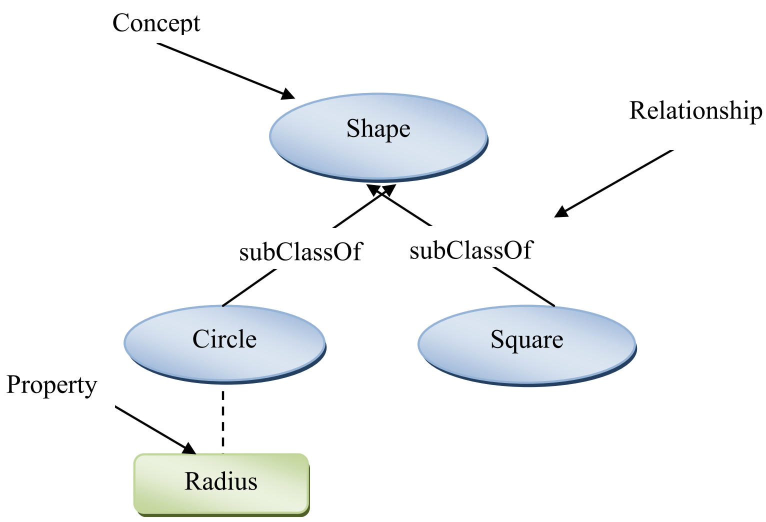

Figure 2.4 Simple ontology

Ontologies provide the primary structure for appending semantics to web services. Modelling with ontologies can be similar to domain modelling in software engineering, and these two disciplines have several similarities, such as inheritance, classes and properties. Ontologies are mainly used in knowledge engineering, knowledge representation and artificial intelligence. However, software developers must adopt ontologies to their standard tools and techniques.

2.2.2 Semantic Web

The semantic web is just an extension of the current web; however, for machine-readability, the semantic descriptions of resources are included. In the semantic web, semantic markup is created for each resource and can be searched and processed using computers with automating tasks that are currently being performed manually [24].

Suppose that a person wants to book a flight, reserve the hotel and rent a car for travelling. In this situation, the person would search the websites to find the required services. For this purpose, there are some constraints, such as date, cost and locality. If all of the competing services were described semantically, the purchaser could express their constraints to a computer program that would be able to make decisions for them. Therefore, there is no need to express and use the exact word or phrase to find appropriate results if all of the services involved have semantic descriptions.

2.2.2.1 RDF

To model information regarding resources on the World Wide Web, Resource Description Framework (RDF) is used as a language. The RDF model is comprised of three elements: the subject, object and predicate [25]. The RDF model is shown in Figure 2.5.

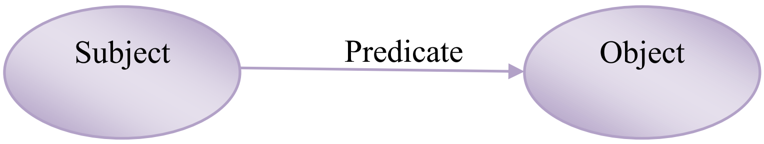

Figure 2.5 Simple RDF model

The *subject* is the “resource” being described and can be anything that has a URI, such as “http://www.public.asu.edu/area51”. The *predicate* is a property of the resource, such as “author”. The *object* is the value of the property, such as “Mojtaba”. Objects can be other subjects or literal values. From this simple model, a directed graph structure emerges. This directed graph structure allows data to be “connected” regardless of how the data are distributed. RDF has a variety of syntaxes, the most popular of which is XML. An example of RDF using the XML syntax is provided in Figure 2.6 [25].

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ex="http://example.org/">
 <rdf:Description rdf:about="http://www.public.asu.edu/~area51">
  <ex:author>John T.E. Timm</ex:author>
 </rdf:Description>
</rdf:RDF>
```

Figure 2.6 RDF example in XML format

2.2.2.2 OWL

Web Ontology Language (OWL) is an XML-based language for describing ontologies [26]. To specify the semantic description on the web, OWL is designed. OWL could be construed by independent software agents. The OWL syntax is platform independent and could thus be easily transferred over a network and manipulated with existing tools. Therefore, the resources can be complex web services or simple web pages. Simple ontology written in OWL is shown in Figure 2.7 [26]:

```
<owl:Class rdf:ID="Shape"/>
<owl:Class rdf:ID="Circle">
  <rdfs:subClassOf rdf:resource="#Shape"/>
</owl:Class>
<owl:DatatypeProperty rdf:ID="radius">
  <rdfs:domain rdf:resource="#Circle"/>
  <rdfs:range rdf:resource="&xsd;integer"/>
</owl:DatatypeProperty>
<owl:Class rdf:ID="Square">
  <rdfs:subClassOf rdf:resource="#Shape"/>
</owl:Class>
```

Figure 2.7 Excerpt of a simple ontology written in OWL

2.2.3 Semantic Web Services

Once the web services are described semantically, these types of web services are termed "semantic web services". In semantic web services, ontologies are used to describe concepts. For such descriptions, semantic languages, such as RDF and OWL, are employed. Ontologies are used to define the meaning of a service and its components. For example, if a web service has an input parameter named "Zip", it would be beneficial to demonstrate that "Zip" is the same thing as "zip code" and "postal code" for an ontology that describes addresses so that those searching for the service would obtain the same result with any of these terms. The most popular state-of-the-art approaches in semantic web service are described Section 2.5.4.

2.3 MCDM

This section discusses one of the significant issues related to service selection. Although the majority of service selection techniques apply, the behaviour of QoS-based service selection allows researchers to resolve the service selection problem using Multi-Criteria Decision Making (MCDM). MCDM consists of two main groups:

> ***MODM***: the decision alternatives are not given. Instead, the set of decision alternatives is explicitly defined by constraints using multiple objective programming. The number of potential decision alternatives may be large [27].
>
> ***MADM***: the selection of an optimal option from among two or more alternative materials based on two or more attributes is a multiple attribute decision making (MADM) problem. These problems are assumed to have a predetermined and limited number of decision alternatives. Some examples of MADM methods are Technique of Order Preference by Similarity to Ideal

Solution (TOPSIS), Resenje (VIKOR) and Analytical Hierarchy Process AHP [27].

MCDM typically considers problems of the following type. Someone wishes to buy one new product and has four aspects in mind that will govern the purchasing choice. Competing manufacturers of that product have offered three options: X, Y and Z. X, Y and Z will all satisfy the requirements, but the MCDM methods select the best choice through the numerous alternatives.

Based on the above definitions, this research is similar to MADM problems. Therefore, in this section, AHP, TOPSIS and VIKOR are discussed.

2.3.1 AHP

AHP is attributed to Saaty [28] and is often referred to as the Saaty method. AHP is a process for developing a numerical score to rank each decision alternative based on how well each alternative meets the decision maker (DM)'s criteria [29].

2.3.1.1 The Basic Principles

A scale should be provided to compare two criteria. For example, to compare price with quality, there should be a scale table indicating which criterion is important. Each of these judgments is assigned a number on a scale [30]. One common scale is shown in Table 2.1. These pair-wise comparisons are carried out for all factors to be considered. The criterion that has the higher preference level will be assigned the number mentioned in the table, and the other will be assigned the reciprocal of the value.

Table 2.1 The Saaty rating scale [28]

NUMERICAL VALUE	PREFERENCE LEVEL
1	*Equally preferred*
2	*Equally to moderately preferred*
3	*Moderately preferred*
4	*Moderately to strongly preferred*
5	*Strongly preferred*
6	*Strongly to very strongly preferred*
7	*Very strongly preferred*
8	*Very strongly to extremely preferred*
9	*Extremely preferred*

A matrix should be created to evaluate criteria weights. Based on the definition, the sample of the matrix for three criteria is shown in Figure 2.8.

Figure 2.8 Matrix for evaluating criteria weights

2.3.1.2 AHP Procedure

This subsection briefly describes the three steps of the AHP procedure. First, the criteria weights are determined. Second, the alternative ratings are computed. Third, the weighted average ratings for each decision alternative are calculated. Then, the alternative with the highest score is chosen. These steps are described in detail below[31].

Step 1:

1) Provide a single pair-wise comparison matrix for the criteria.
2) Sum the values in each column.
3) Normalise the data by dividing the above sum of columns to obtain suitable weights.
4) Sum the values in each row and calculate the average to determine the final weights.

Step 2:

1) Develop a pair-wise comparison matrix for each criterion, with each matrix containing the pair-wise comparisons of the performance of decision alternatives on each criterion.
2) Repeat stages 2 to 4 of *step 1* for each alternative.

Step 3: Calculate the weighted average rating for each decision alternative and select the alternative with the highest score.

2.3.2 TOPSIS

TOPSIS selects the alternative that is the closest to the ideal solution and farthest from the negative ideal alternative [32].

Ideal alternative: the alternative that has the best level for all attributes considered.

Negative ideal alternative: the alternative that has the worst attribute values.

TOPSIS assumes that there are *m* alternatives (options) and *n* criteria (attributes) and that there are the scores of each option with respect to each criterion.

2.3.2.1 TOPSIS Procedure

The TOPSIS procedure consists of the following five steps [33]:

Step 1: Transform various attribute dimensions into non-dimensional attributes in a normalised decision matrix. The normalisation of the TOPSIS method is as follows:

$$r_{ij} = \frac{x_{ij}}{(\sum_i x_{ij}^2)} \; for \; i = 1, \dots m; j = 1, \dots, n \quad (1)$$

Step 2: Construct the weighted normalised decision matrix. Multiply each column of the normalised decision matrix by the associated weight. The criteria weights are shown by w_j, for j = 1,…n:

$$v_{ij} = w_j r_{ij} \quad (2)$$

Step 3: Determine the ideal and negative ideal solutions.

Ideal solution: $A^* = \{v_1^*, \dots, v_n^*\} \; where$

$$v_j^* = \{\max_i(v_{ij}) \; if j \in J; \; \min_i(v_{ij}) \; if j \in J'\} \quad (3)$$

Negative ideal solution: $A^* = \{v_1', \dots, v_n'\} \; where$

$$v_j' = \{\min_i(v_{ij}) \; if j \in J; \; \max_i(v_{ij}) \; if j \in J'\} \quad (4)$$

Step 4: Calculate the separation measures for each alternative. The separation from the ideal alternative is

$$S_j^* = \left[\sum_j \left(v_j^* - v_{ij}\right)^2\right]^{1/2} i = 1, \dots, m \tag{5}$$

Similarly, the separation from the negative ideal alternative is

$$S_j^{'} = \left[\sum_j \left(v_j^{'} - v_{ij}\right)^2\right]^{1/2} i = 1, \dots, m \tag{6}$$

Step 5: Calculate the relative closeness to the ideal solution C_i^*:

$$C_i^* = S_i^* / (S_i^* + s_i^{'}) \;\; 0 < C_i^* < 1 \tag{7}$$

The option with C_i^* closest to one is the best selection.

2.3.3 VIKOR

VIKOR is a multi-criteria optimisation method for complex systems. "It determines the compromise ranking-list, the compromise solution, and the weight stability intervals for preference stability of the compromise solution obtained with the initial (given) weights" [34]. The goal of this method is to rank and select a set of alternatives with conflicting criteria. VIKOR addresses the multi-criteria ranking index based on the particular measure of "closeness" to the "ideal" solution [35]. The compromise solution (F^c) is a feasible solution that is the "closest" to the ideal solution (F^*) [35]. The compromise and ideal solutions are presented in Figure 2.9.

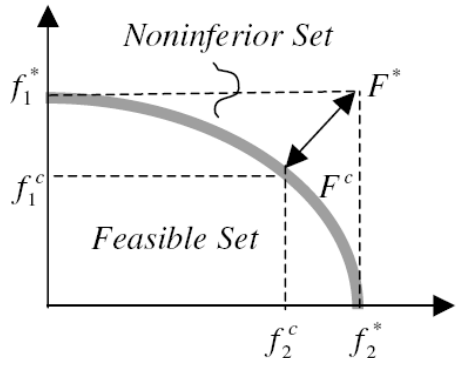

Figure 2.9 Ideal and compromise solutions

2.3.3.1 VIKOR Procedure

VIKOR is a method suitable for problems having numerous alternatives [34], similar to service selection problems in which there are many available services. To propose the service selection method, assume that there are m alternative services $(A_1, A_2, A_3, \dots, A_m)$ with respect to n QoS $(C_1, C_2, C_3, \dots, C_n)$. The steps of the VIKOR method for service selection are described below.

Step 1. Determine f_j^* and f_j^-, which are the best and worst values of each criterion, respectively, where *j=1,2,...,n,* and specify the maximum value and minimum values of each column in the decision matrix. The maximum and minimum values represent the highest and lowest values for the benefit criterion and lowest and highest values for the cost criterion, respectively.

Step 2. As the scales of each criterion are not equivalent, the decision matrix should be normalised. The dimensions of "performance" and "price" are on different scales. Thus, the VIKOR method uses linear normalisation so that the result remain unaffected when the scale of the criteria changes. S_i and R_i are formulated as follows:

$$S_i = \sum_{j=1}^{n} W_j \left(\frac{f_j^* - f_{ij}}{f_j^* - f_j^-}\right) \tag{8}$$

and

$$R_i = \max_j \left[W_j \left(\frac{f_j^* - f_{ij}}{f_j^* - f_j^-}\right)\right] \tag{9}$$

where $f_{ij}\ (i = 1,2,3,\ldots,m\ and\ j = 1,2,3,\ldots,n)$ and $\mathrm{X_{ij}}\,(\mathrm{i} = 1,2,3,\ldots,\mathrm{mandj} = 1,2,3,\ldots,\mathrm{n})$ are the elements of the decision matrix. Alternative *i* with respect to criteria *j* and $\mathrm{W_j}$ represent the importance weights of the criteria.

Step 3. Compute the index values. These index values are defined as

$$\boldsymbol{Q_i} = \begin{cases} \left[\frac{R_i - R^-}{R^+ - R^-}\right] & if\ S^+ = S^- \\ \left[\frac{S_i - S^-}{S^+ - S^-}\right] & if\ R^+ = R^- \\ \left[\frac{S_i - S^-}{S^+ - S^-}\right] v + \left[\frac{R_i - R^-}{R^+ - R^-}\right](1 - v) & otherwise \end{cases} \tag{10}$$

S^-, S^+, R^-, and R^+ are defined as

$$S^- = Min\ S_i\ ,\ S^+ = Max\ S_i \tag{11}$$

and

$$R^- = Min\ R_i, R^+ = Max\ R_i \tag{12}$$

The value of *v* is introduced as a weight for “the majority of criteria” strategy (or “the maximum group utility”), where$1 - v$ is the weight of the individual regret.

The value of *v* ranges from zero to one, and a compromise between these strategies can be reached by using *v*=0.5.

Step 4. Sort the values S, R and Q in decreasing order to obtain three ranking lists.

Step 5. Propose as a compromise solution for the alternative with the highest ranking $(\mathrm{A}^{(1)})$using measure Q (minimum) if the following two conditions are satisfied:

C1. Acceptable advantage:

$$\mathrm{Q}(\mathrm{A}^{(2)}) - \mathrm{Q}(\mathrm{A}^{(1)}) \geq \mathrm{DQ} \tag{13}$$

where $A^{(2)}$ is the alternative with the second-highest ranking according to Q and DQ is calculated in the formula below, where *M* is the number of alternatives.

$$DQ = \frac{1}{(M-1)} \tag{14}$$

C2. Acceptable stability in decision making:

The alternative $\mathrm{A}^{(1)}$ should also be the ranked the highest by *S*and/or*R*.

A set of compromise solutions is proposed as follows if one of the conditions is not satisfied.

Use alternatives $A^{(1)}$ and $\mathrm{A}^{(2)}$ if only*C2* is not satisfied or alternatives $\mathrm{A}^{(1)}A^{(1)}$,$A^{(2)}$,…,$A^{(M)}$ if*C1* is not satisfied; $\mathrm{A}^{(\mathrm{M})}$is determined by the following relation for maximum *M*:

$$Q\left(A^{(M)}\right) - Q\left(A^{(1)}\right) < DQ \quad (15)$$

The alternative that has a minimum Q is the best alternative. The core ranking result is the compromise ranking list of alternatives and the compromise solution.

2.4 Web Service Selection (WSS)

Based on a recent research project, the web service mechanism is separated into discovery, selection and composition [7]. Web service discovery allows providers to publish service descriptions and profile information regarding businesses and services and other related details in UDDI repositories. However, there are instances in which there is a need to utilise non-functional properties and select the most appropriate service to cater to user requirements, apart from functional properties. The selection component is used for this purpose. Finally, web service composition combines the selected services within the time frame required. A set of services can be composed as a composite service to provide the requisite functions [8].

When more than one web service meets the functional requirements, the WSS uses some criteria to select the best candidate service [36]. The value of non-functional properties in these matching web services may be different, but they should meet certain minimum requirements. The selection criteria may have an interdependent relationship. The number of methods used for decision making is considered in WSS because of the complication that exists during the selection process[37].

WSS occurs when there is a set of discovered web services that can fulfil the user's requirements and one of these services should be selected to return back to the service consumer [10]. This selection must be tailored to the user preferences; whereas one user may prefer a high-quality alternative, another may prefer a less expensive alternative[11]. The value of non-functional properties in these matching web services may be different, but they should have certain minimum requirements. Non-functional properties of service descriptions can be expressed semantically by logical rules using terms from non-functional property ontologies [4]. The discovery and selection of services are shown in Figure 2.10.

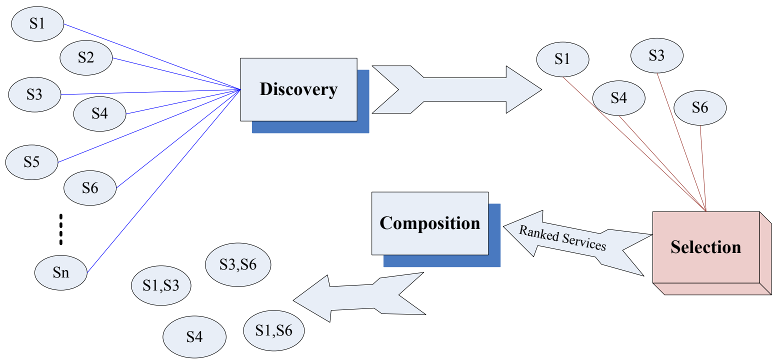

Figure 2.10 Discovery, selection and composition of services

Two significant tasks in the process of using services are selection and ranking, and every solution for these tasks is affected directly by the description of services. When describing a service, three items must be considered: behaviour description, functional description and non-functional description.

The formal specification of how the functionality of the service can be achieved is described in the behaviour description. The formal specification of the exact ability of the service is described in the functional description. Capturing constraints on the previous two specifications is described in the non-functional description.

These three descriptions are extremely important for many service-related tasks. When selecting the most appropriate service among a sorted set of services, non-functional properties are essential input data that must be considered [4].

2.4.1 General Framework for WSS

The three parts of the traditional service model are: the service consumer, SP and UDDI registry. The framework below extends the overall architecture proposed to support service selection, ranking and quality updating, which consists of the web service repository, service selection module, QoS database, SP, service requester and quality rating database. The general framework for WSS with QoS is shown in Figure 2.11 [38].

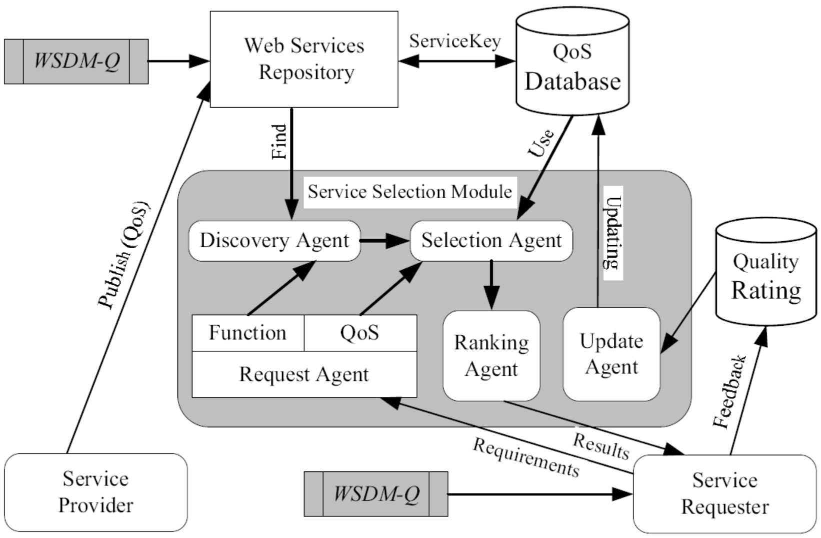

Figure 2.11 General framework of WSS

As highlighted in the figure, the core of this framework is the service selection module. This module contains the following five main agent components [38]:

- The discovery agent is responsible for finding the initial web service set that satisfies the service requester's functional requirements.

- The selection agent collects QoS information from the QoS database in terms of the initial discovered web service set and then selects the web service set that fulfils the service requester's QoS limitations.

- The request agent provides the interface and communicates with the service requester to obtain functional requirements and QoS limitations.

- The rank agent calculates the QoS score for each selected web service and then ranks these services in decreasing order according to their QoS marks. Finally, the ranked service set is returned to the service requester.

- The update agent modifies the quality criteria value in the QoS database according to the feedback information in the quality rating database.

For the remaining components in this framework, the registry mechanism is provided by a web service repository for the web SPs that publish functional and non-functional properties. In particular, published QoS information is stored in the QoS database correlative with the web service repository by the ServiceKey. All of the QoS feedback information from the service requesters' invoking services is collected by the quality rating database.

2.4.2 QoS for WSS

According to [39], "QoS will become a significant factor in distinguishing the success of SPs". Some have suggested that in the evaluation of web services, QoS issues should be from the perspective of both the users of the web services and the providers of these services [40].

Currently, service discovery and selection are performed manually, but if numerous services are available for selection, these tasks should be performed automatically[41].

Occasionally, there are more than one registered services with similar functionality. In such a situation, the QoS-based service selection plays an important role in the matchmaking process. The QoS attributes can be used for matching as non-functional properties. These attributes are described below[42].

In [19], some criteria are defined for the goal component of WSMO. Therefore, the QoS weights based on those criteria are defined, which are measurable and needed for service selection. Those criteria are as follows[19]:

- ✓ *Performance (P)* is how fast a request of service can be completed, which is an essential element for web services. The waiting time and execution time are required to estimate P. The waiting time is the duration for activities, such as transferring a message, and the execution time is the duration of performing the functionality of a service [43-44].

- ✓ *Accuracy (A) is* how accurate the service result is or the rate of error generated by the service; in fact, this is different from the approach accuracy.

- ✓ *Scalability (S)* is the capability of the service to process more requests in a certain amount of time. If the scalability of a service is high, the performance and accuracy of the service are not affected by an increased number of requests.

- ✓ *Financial (F)*, also called *Cost* or *Price*, concerns the cost and charges related to a service [45]. The cost of requesting and using each service is the web service price. The price of services is affected by the functionality value. Providing more complex functions increases the cost of the service.

- ✓ *Reliability (R)* is the ability of a service to achieve its requested tasks and functions. The capability of the SP to deliver requested service functionality successfully is web service reliability. The probability of success in a service execution defines the quantity of this capability. However, the failure rate of a service typically determines the reliability. The rate is evaluated as the ratio of the execution time to the mean time between failures. The execution time can be in conflict because it is the time required to perform a service and also the time required to deliver a result from the service requester's perspective; however, because the SP is not able to support the network problem, execution time is considered as the time required to perform a service.

- ✓ *Trust (T)* is the trustworthiness of the service, which is in contrast to reputation, which relates to the recognition of the SP.

- ✓ *Security (Se)* is the ability of a service to provide authorisation, authentication, data encryption and traceability.

2.5 Approaches in WSS

In recent years, there has been increasing interest in service selection based on QoS. Thus, this research focuses on the approaches based on QoS. To classify the approaches in service selection, three essential aspects should be considered:

- ❖ How should the QoS for capturing preferences be modelled?
- ❖ How should the QoS data be gathered?
- ❖ How should the decision be made (the algorithmic aspect)?

There are two methods for answering the first question (improve protocol and semantic), second question (policy and trust/reputation) and third question (standard MCDM methods and atypical decision-making methods). Based on the above questions, the proposed WSS classification is shown in Figure 2.12.

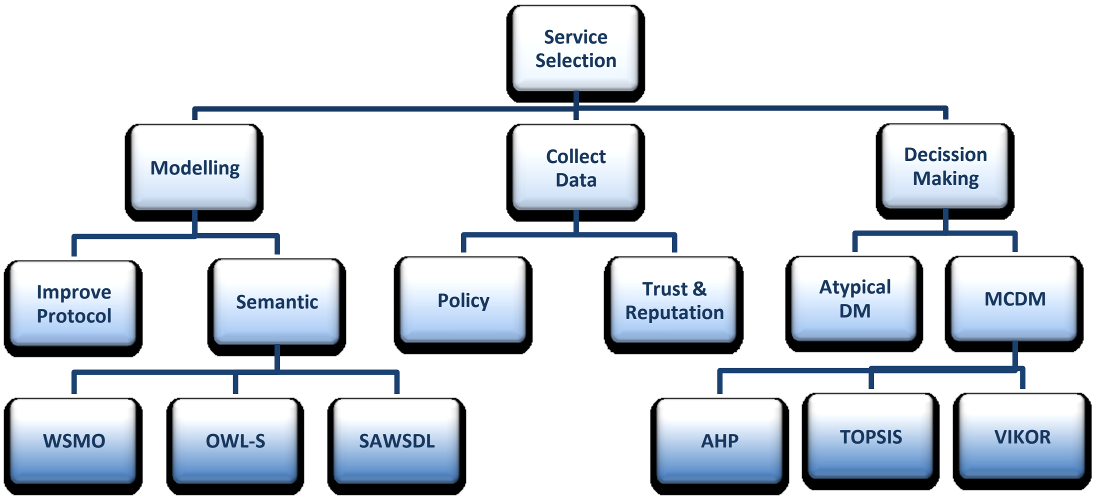

Figure 2.12 WSS classification

This classification includes two levels under the main categories. The first level includes improve protocol, semantic, policy, trust and reputation, atypical DM and MCDM. The second level involves two sub categories: Semantic and MCDM. The semantic level includes WSMO, OWL-S and SAWSDL while the MCDM level includes AHP, TOPSIS and VIKOR.

2.5.1 Policy-based Approach

This approach is based on policy languages, and only a limited number of non-functional properties are accepted. The QoS policy model is used to define the non-functional requirements. Some examples of policy-based approaches include those proposed by[9] and [46].

The policy-based designs model the QoS policy as a textual document. Preferences and non-functional limitations of the service requestor are shown in the content of the policy model. A matrix is used to present the non-functional criteria's relations and is also applied for their aggregation [47].

The human involvement required by this approach is one of its disadvantages. Validating the non-functional criteria by evaluating the overall score is difficult. Furthermore, once the model wants to present requirements, how and where the values of properties are stored is not important. It is difficult for users to understand the matrix aggregation function because of its complexity. In addition, the satisfaction level of the service is not considered in the final ranking scores that are expected by the service[47].

2.5.2 Trust and Reputation-based Approach

This approach relies on feedback from other users who have used the service. In this situation, each service can be popular if it has a good reputation. In contrast to the policy-based approach, which is based on the informed quality of the suggested service, this approach is based on the SP's standing. Some approaches are presented in [48], [49] and [14] based on trust and reputation. In this category, the approaches are divided into those relying on views of the services before making decisions and those capturing user requirements by policies.

The criteria of WSS described in [48] are determined for all types of services in a static manner. Each trust and reputation typology is related to a criterion. For different services, the criteria's values are obtained through the typology according to feedback from agencies and communities. One of the selection functions is PageRank [50],which is used by a centralised reputation system. A similar idea that uses the IRS-III [51] selection methodology based on ontology mapping technology to calculate the ranking scores is presented in [49]. In all of these approaches, trust is a fundamental item for evaluating selection criteria, and a trusted party provides the values for the services. These approaches have some common disadvantages [47]:

- A model of expressing service properties is not determined, and these approaches suppose that the service properties values can be obtained easily.

- These approaches only use a function to evaluate every non-functional property, and the evaluation of functions in each situation is not addressed.
- These approaches do not consider all of the specifics for aggregation functions.

2.5.3 Improve Protocol-based Approach

Solutions include adding new actions to standard UDDI to achieve a dynamic UDDI process and designing a selecting language, such as SQL, and selecting web services by setting the limiting situation [52].

This research focuses on UDDI extension, which is the most popular solution. Examples of approaches based on UDDI extensions for service selection can be found in[53] and [54].

The approach in[53] adds *Quality broker* as an additional component in the SOA, which is located between the UDDI and repository service requestor. In this approach, only three non-functional properties are addressed: safety, performance and cost. Thus, additional non-functional properties cannot be added. Furthermore, the level of matching will be expressed as three types of values: gold, silver and bronze. The main disadvantages of approaches based on UDDI are as follows[47]:

- Separating service data and quality information causes the registered services to be monitored dynamically by the quality broker or the service's details to be stored in more than one location by the SP.

- There are only a few approaches for selections based on the limited number of criteria because the service quality model is not extensible.

2.5.4 Semantic-based Approach

In some approaches, semantic web service is addressed to define non-functional models for selecting web services. There are some approaches for semantic WSS, such as WSMO, WSDL-S, DAML-S, SAWSDL and OWL-S. Similar to SAWSDL, OWL-S is the new version of WSDL-S and DAML-S. Thus, this research focuses on three approaches that have relatively good reputations: WSMO, OWL-S and SAWSDL.

2.5.4.1 WSMO

Web Service Modelling Ontology (WSMO) is a group of modelling elements used to describe all aspects of semantic web services [55]. Four main parts of WSMO are *web services*, *ontologies*, *goals* and *mediators*. WSMO's elements are shown below [19].

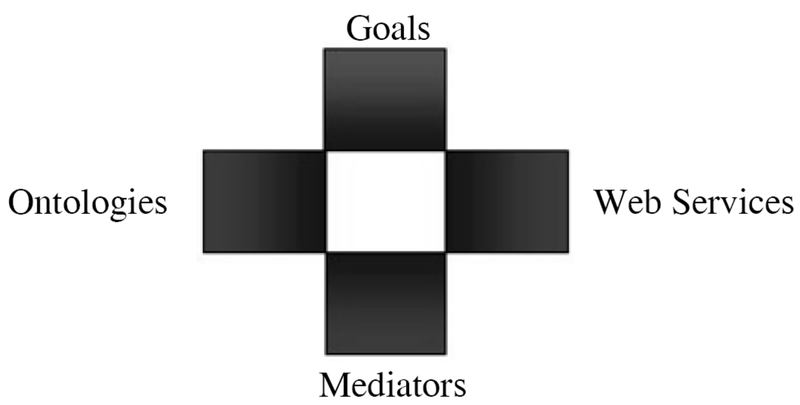

Figure 2.13 Core WSMO elements

Ontologies are the key WSMO component used to describe domain concepts. The descriptions of the service and how to access it are web services. A *web service* in WSMO attends an atomic unit of functionality that is accessed through standard XML-based messaging protocols. *Goals* are descriptions of what the end user wants to achieve. Goals are targets that are specified by the client when the client wants to use the service. *Mediators* are descriptions of semantic incongruities between other WSMO elements and are used for solving interoperability issues. Between resources in a WSMO description, there may be heterogeneity, and mediators can address conflicts at both the process level and data level [56].

❖ WSML

WSML is a language use to model web services, ontologies and related aspects and is based on WSMO [57]. WSML is based on different logical formalisms, namely, description logics, first-order logic and logic programming. WSML consists of a number of variants based on these different logical formalisms, namely, WSML-Core, WSML-Rule, WSML-Flight, WSML-DL and WSML-Full. WSML-Core corresponds with the intersection of description logic and Horn logic. The other WSML variants provide increasing expressiveness in the direction of description logics and logic programming. Finally, both paradigms are unified in WSML-Full, the most expressive WSML variant [58]. Figure 2.14 presents the different variants of WSML.

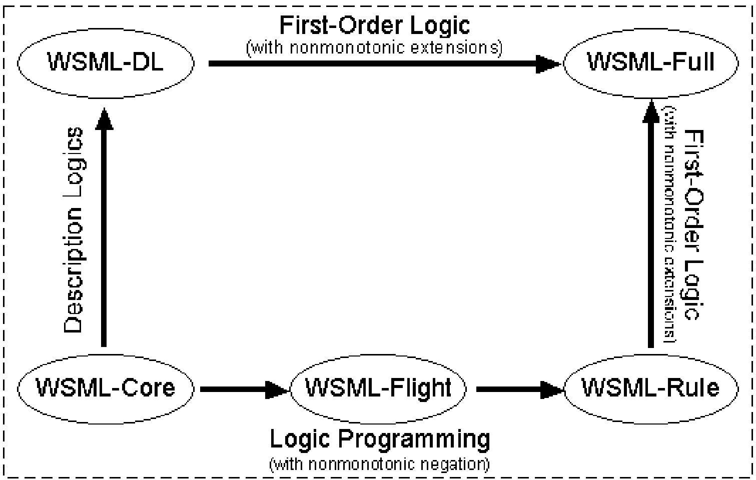

Figure 2.14 Type of WSML

In this type of service selection approach, the criteria of services as elements of the selection process are described by non-functional properties. Suitable modelling properties must be provided and then added to the goals and services. Thus, the QoS is the most relevant approach for selecting service and ranking tasks [4].

- **Ranking Solution**

As discussed in [59], ranking approaches can be categorises as *global* or *local* and *absolute* or *relative*. Google's PageRank approach fits in the absolute and global category, whereas the ranking mechanism for services is categorised as relative and local.

The algorithm for the ranking uses multi-criteria non-functional properties. This algorithm considers some elements, such as the importance of the criteria, which has a value between zero and one and a default value of 0.5. This element indicates the importance of the specified criteria from the user's perspective. If the value of importance is one, then the user is interested in the non-functional properties. A value of zero indicates that the user is not interested in the non-functional properties. In addition, a tuple is considered for holding non-functional properties and their related importance, and a quadruple set is used to retain the services that can satisfy the user requirements. The non-functional properties and their importance are saved for each quadruple record. Finally, the type of ordering can be determined for a specified variable.

Once the above conditions are established, the system should verify that each service can satisfy user requirements and that the non-functional properties of each service appear in the goal description. The specified rule is extracted and then evaluated, and the specified non-functional properties of the goal are provided in the service description. After the evaluation of the rule, the declared quadruple can be filled with the required data. The score of each non-functional property is collected. Then, by adding these data, the final scores for each service are estimated. Finally, by referring to the value of the ordering variable, the available list can be sorted, and the sorted list returns to the selection step [4].

- **Selection**

The last step is selection, which can be performed after the ranking process is complete. In this step, the best candidate service is selected. This process is significantly easier than service ranking. The top *k* services returned from the sorted list in the ranking process are selected. The user specifies the value of *k*in the goal description using an annotation.

2.5.4.2 OWL-S

Web Ontology Language for Web Services (OWL-S) is one of the semantic web service frameworks that supports the semantic representation of services with a close connection to OWL. OWL supports reasoning's subsumption on the classifications of concepts. Furthermore, in concepts, definition relations are simplified by OWL [60]. The component of OWL-S can be described by the answers to three questions: "what does the service do?" (*ServiceProfile),* "how is the service accessed?" (*ServiceGrounding*) and "how does the service work?"(*ServiceModel*). The *ServiceProfile*is initially used for discovery, whereas the *ServiceGrounding* and *ServiceModel* are used in selection and invocation[15]. Figure 2.15 shows the three-part division of the OWL-S ontology [25].

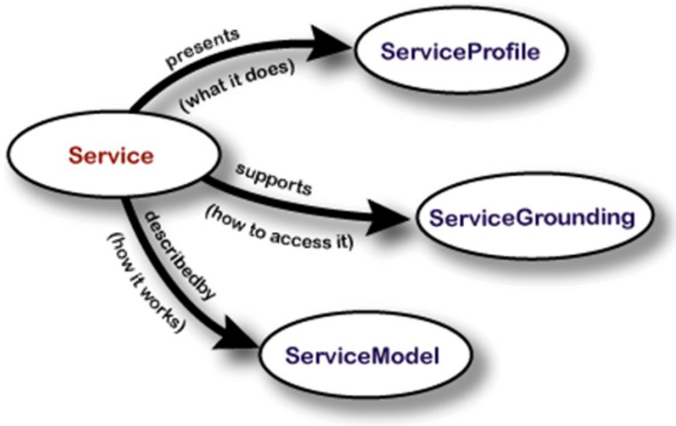

Figure 2.15 OWL-S Conceptual Model

The OWL document is used to consider the semantic WSS so that the non-functional and functional requirements can be specified [61]. This document has a number of memory locations for storing web services descriptions. For describing the services, WSDL is applied to find a matching service using an OWL description of the user requirements [12]. In this approach, the user's view is addressed in the web service composition. For better quality, the service is considered in a composition [62].

In every composition step, a number of services or composite services that have higher reputations are used for the composition. The user experience is used in this approach. The semantic matcher and matching algorithm are described below[62].

- **Semantic Matcher**

The semantic matcher architecture is shown in Figure 2.16. The web service descriptions are analysed by the parser to establish different parameters to be matched.

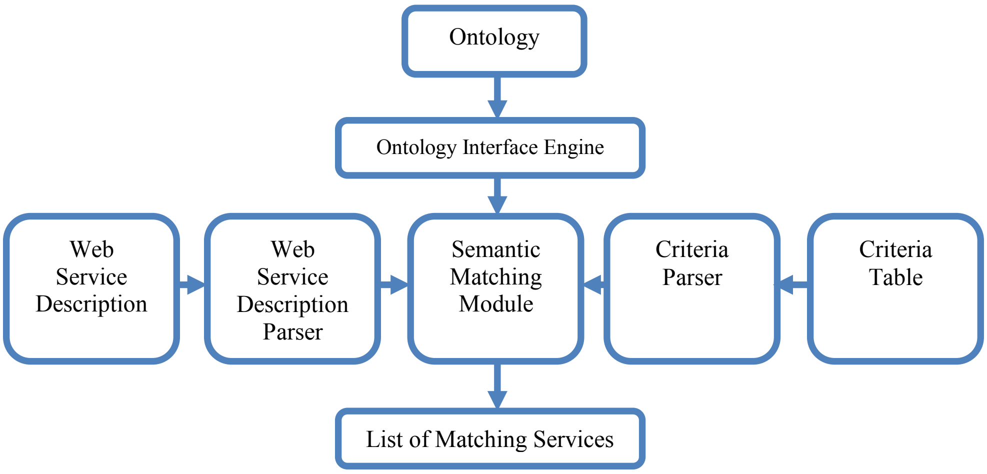

Figure 2.16 Architecture of the semantic matcher

The types of matching in this approach are as follows:

- ✓ ***Exact***: Both conceptual annotations are identical.
- ✓ ***Plug-in/subsumption:*** Plug-in is second conceptual annotation concentrates on the first annotation and subsumption is vice versa.
- ✓ ***Container/part of***: The container is first annotation contains the second annotation and vice versa is the part of.
- ✓ ***Disjoint***: Otherwise, the annotations are disjoint.

❖ **Matching Algorithm**

The matching algorithm is described below[63].

Service: A set of features that describes the service is defined as S.

Similarity Measure: The mapping used to measure the semantic space in the middle of two conceptual annotations is a similarity measure that is represented by μ.

$$\mu: A \times A \rightarrow \{Exact, Plugin, Subsumption\ , Container, part\ of, disjoint\}$$
$$\text{Exact} \succ \text{Plug in} \succ \text{Subsumption} \succ \text{Container} \succ \text{part of} \succ \text{Disjoint}$$

Formally (1),

$$\forall_i \exists_j (S^R.A_i = S^A.A_j) \wedge \mu(S^R.A_i, S^A.A_j) \succcurlyeq \mu_i \Rightarrow SuffMatch(S^R, S^A)\ 1 \leq i \leq S^R.N$$

C is the relation between two attributes $C.A$ and $C.\mu$, where $C.A$ is the service attributes that are to be compared and $C.\mu$ is the minimum preferred similarity measure for the specified attribute.

A sufficient match exists between $S^R$and S^A if, for each attribute in $C.A$, there is an equal attribute of $S^R$and $S^A$and the defined similarity measure in $C.\mu$ can be satisfied by the values of the attribute.

(2):

$$\forall_i \exists_{j,k} (C.A_i = S^R.A_j = S^A.A_k) \wedge \mu(S^R.A_j, S^A.A_k) \succcurlyeq C.\mu_i \Rightarrow SuffMatch(S^R, S^A)\ 1 \leq i \leq C.N$$

2.5.4.3 SAWSDL

SAWSDL is semantic annotations for WSDL. To enrich description of services and their semantic annotation, SAWSDL is used as an expansion of WSDL [16]. For this purpose, XML attributes that include the notion of schema mapping and model reference are added to the current WSDL elements. These additional elements are presented in Figure 2.17 [13].

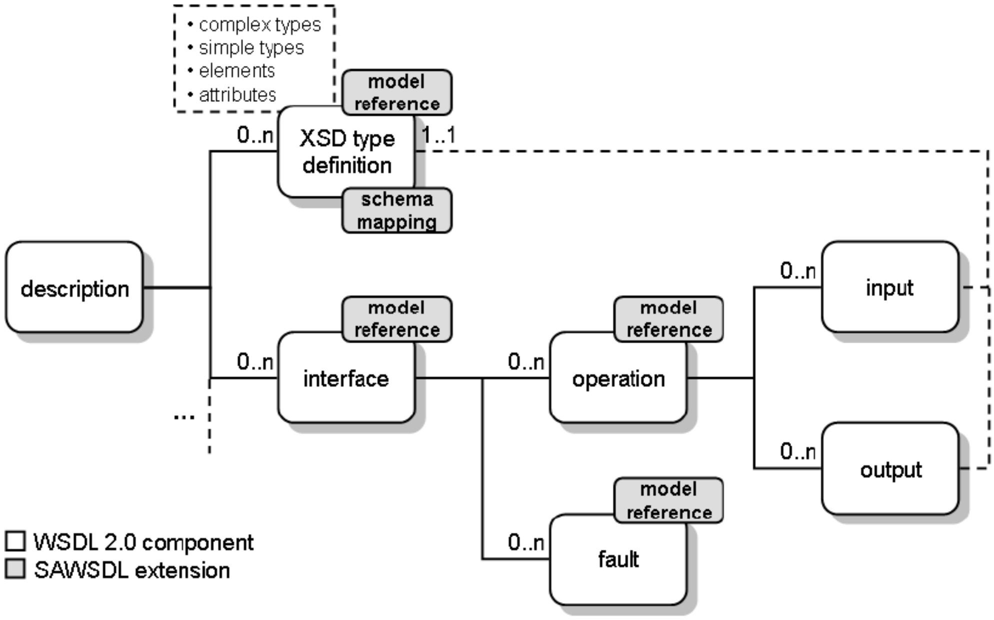

Figure 2.17 Interface components of SAWSDL

Schema mappings are candidates for sustaining the automatic service execution. Schema mappings provide rules of communications between the lower and upper levels. The lower level refers to the XML schema, and the upper level refers to semantic annotation in the specified ontology [64].

ModelReference refers to concepts with similar candidate meanings expressed in an optional semantic language that can define each element of the XML schema. However, the WSDL operations, simple types, complex types, interfaces and attributes are defined by the SAWSDL specifications. The supporting automatic discovery of service is the essential purpose of a model reference.

liftingSchemaMapping expresses the transformation from the lower level to the upper level.

LoweringSchemaMapping expresses the transformation from the upper level to the lower level.

One type of approach in semantic WSS is called SAWSDL-MX, which performs the service selection in terms of logic-based, syntactic and hybrid matching of I/O parameters defined for potentially multiple operations of a web service interface. Standard SAWSDL web service definition documents are used as service requests [13].

The matching process of SAWSDL-MX on the service interface level is performed as follows. For each service request (R) and service offer (O) pair, every combination of its operations is evaluated using logic-based matching, text retrieval-based matching or both.

SAWSDL-MX uses two fold graph matching to compute an optimal mapping of operations for the service request and offer. The nodes of the graph signify the operations, and the weighted edges are built from possible one-to-one assignments with their weights derived from the computed degree of operation match. The service offer provides an operation when there is any available mapping, indicating that there is no request operation that cannot be provided by the service offer, ignoring the match quality at this position [13].

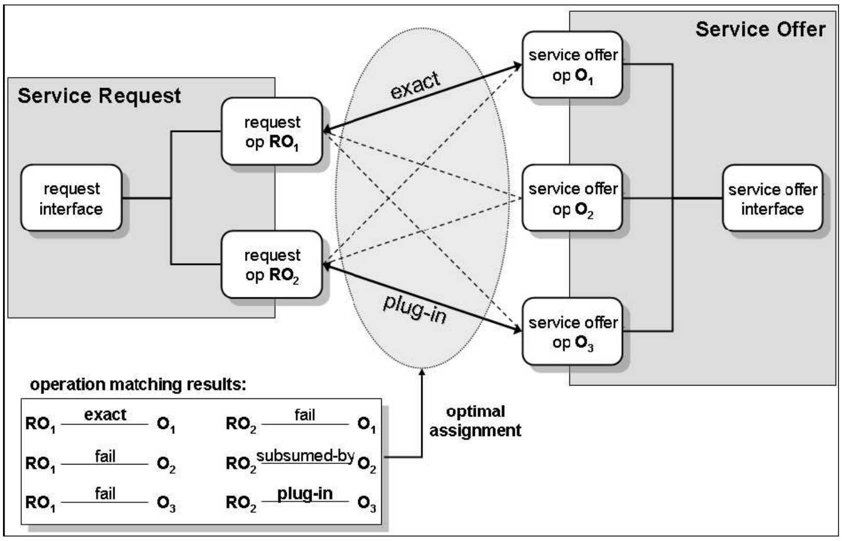

Figure 2.18 Matching level interface of SAWSDL-MX

There is a cautious option for defining the matching level between the service request and offer that is supposed to be the worst result in the best operation matching. A cautious option an assure a similarity fixed lower bound for each required operation, which is the task of SAWSDL-MX, as shown in Figure 2.18. For the request, the service offer is considered as a plug-in match. The other alternative is to merge the results of the operation matching derived from their average syntactic similarity values.

The algorithm for logic-based matching of the operations uses the filters in the following order based on the level of relaxation:

$$exact > plug-in > subsumes > subsumed-by > fail$$

2.5.5 Decision Making

As described in Section 2.5, one of the aspects of WSS is decision making. There are two types of the decision making: *MCDM* and *atypical decision making*. In the former, MCDM methods, such as AHP, ANP, TOPSIS, linear programming and fuzzy programming, are employed. In the latter, just the naive methods are used. The accuracy of the former approaches is higher than that of the latter. The accuracy, precision and performance of MCDM methods are already proven, whereas the new naive method is not sufficiently reliable to use for the selection of web services. Some of the approaches that use these methods are discussed below.

2.5.5.1 MCDM Approaches

MCDM is one of the approaches that has been used for service selection. The nature of the service selection problem allows MCDM to be used in the algorithm. In this section, several approaches that used MCDM methods in the selection algorithm are investigated.

A linear programming method is considered in [65].The proposed approach is based on using QoS and fuzzy linear programming to discover dissimilarities between service alternatives and to select the most appropriate services with respect to user preferences. In addition, as the approach is an optimal method, the results of the approach are not affected additional criteria in the decision matrix, demonstrating the scalability of the approach.

A fuzzy model is applied in [66] to solve the service selection problems based on QoS. In the proposed method, the weights of QoS criteria could be analysed from the evaluation of existing information. In this approach, customers can obtain a dynamic ranking of the accessible services. Furthermore, a new method for determining the QoS is exploited to select the right service based on the customer's preference.

In [67], the approach developed a general QoS-based service selection method. The authors propose a MCDM method that solves the problem based on TOPSIS with a focus on QoS. The method can declare the non-functional properties of web services by importing the proposed QoS ontology into the OWL-S model. The QoS values of a web service are normalised with higher normalised values corresponding to higher levels of service performance.

In [8, 68], similar approaches using hybrid models based on fuzzy logic and the TOPSIS method are considered. In [68], the fuzzy TOPSIS method is utilised to solve the service selection problem when a group of users have different preferences for the assessment of services. To evaluate the weights of the criteria and the ratings of each alternative web service, the linguistic terms are depicted by triangular fuzzy numbers and then converted into crisp numbers. Finally, to measure the distance of each alternative service from the positive ideal solution (PIS) and the negative ideal solution (NIS), the Minkowski distance function is applied. In [8], an approach based on a new user centric service-oriented modelling is proposed. The proposed method combines the fuzzy TOPSIS method and service component architecture (SCA) to facilitate service development and efficiently satisfy user preferences.

Experiments are also performed on a simulated environment, which includes four 8×8 LED matrices representing 30 services to form 10 composite services for selection. The scalability of the system is also demonstrated; the system remains efficient when the number of services is increased.

In [69], the use of the ANP for WSS is discussed and a network model with a set of relevant components for web services are proposed. The advantages of ANP and how it provides final ranking are considered. In this approach, the decision matrix criteria are prepared with respect to QoS. The ANP is applied for the weighting of criteria, but the user preference is neglected in this approach. This approach is not scalable because the super matrix is composed of several sub-matrices and the size of the super matrix is dependent on the number of criteria.

An enhanced PROMETHEE is proposed in [70] to solve QoS-based service selection. This paper considers the relationship among the QoS criteria; for this purpose, ANP is used to evaluate the criteria weights. There are two types of ranking: one is based on net outranking flows, and the other one is based on outranking flows considering the user requests. In this situation, the overhead of the approach is high, and performance and scalability will be affected.

The approach in [52] focuses on how to solve the service selection problem based on the AHP method. The approach creates an index system for web services selection from four aspects: the user, supply side, product and environment. Then, the visions of 30 professionals are collected using the AHP method. Finally, the weight of each index at all levels based on the data collected from the questionnaire is calculated. Although this approach is based on user preference, QoS is not adequately considered.

2.5.5.2 Atypical Decision Making Approaches

Some WSS approaches use a naive algorithm for selection services. In [71], a service-ranking approach based on semantic descriptions of non-functional properties of services is proposed. In [71], the attachment of non-functional properties to services and goals in WSMO is explained. The proposed ranking mechanism uses logical rules to describe the non-functional properties of services and evaluates the properties using a reasoning engine. Finally, the approach constructs a ranked list of services based on user preferences considering the values calculated in the rule evaluation stage.

2.6 Summary

In this chapter, the definition of the essential concepts of web service, ontology, semantic web, semantic web service and WSS are described. This chapter also investigated the modelling and algorithms of state-of-the-art WSS approaches. To provide WSS, the concept of MCDM is introduced. Moreover, some MCDM methods and their processes are described in detail.

This study indicates that there are certain approaches that focus on modelling, whereas the other approaches focus on the algorithm. However, a few approaches employ both aspects. Although mature approaches attempt to improve the accuracy of the algorithms and models, none have explicitly verified the accuracy of the input data. In the following chapter, an evaluation of the state-of-the-art approaches is presented.

CHAPTER 3

COMPARATIVE EVALUATION OF WSS APPROACHES

This chapter describes a comparative evaluation of the state-the-art approaches that are discussed in Chapter 2. First, the WSS criteria are described. Second, the comparison in the first level of classification is discussed. Third, the comparison of the second level, which is divided into the semantic level and MCDM level, is discussed. Finally, the approaches are evaluated.

3.1 WSS Criteria

In this section, the common criteria for the evaluation of WSS are described. The WSS approaches that are described in Chapter 2 are compared based on the criteria described below.

3.1.1 User Preference

User preferences express how important certain non-functional properties are from the users' perspective[72]. This criterion refers to the approaches that consider user preference to account for the precedence of service consumers. For instance, the relative importance of criteria in a decision matrix can be obtained from the preference of the service requester. Depending on the situation of the service requesters, preferences typically vary for the non-functional criteria, and different requesters have different preferences. Regarding the expression of non-functional properties, the values of criteria be expressed and the relation with other non-functional properties should be considered. For example, when a user requests a financial service, she may consider privacy more important than security.

3.1.2 Performance

Performance refers to output results from processes of service selection, namely, the efficiency of the approach in the selection of the services. In service selection, performance is also termed response time, which includes the duration of running, waiting and executing. The performance scale is inversely related to the execution scale; when the response time is lower, the performance of approach is higher.

3.1.3 Accuracy

Accuracy refers to both *accuracy of data* and *accuracy of method.* Accuracy of data are related to the trustability and reliability of input data. Accuracy of method refers to the precision of the selection method. Although the mature WSS approaches do not address the accuracy of input data, this criterion refers to the accuracy of both the input data and the selection method. The accuracy is typically evaluated using precision and recall [73]. Because selection is the final stage of determining the appropriate service, accuracy is the most important criterion.

3.1.4 Automation

Automation of service selection is one of the challenges in the area of service selection [74]. The final step is essential to automatic service selection. When a service is available, the service designer specifies the data for the service and the user specifies the requirements. However, service selection is performed automatically. The automation in service selection is referred to as the process of ranking and selection.

3.1.5 Scalability

Scalability refers to the approaches that consider numerous properties and ranking processes that occur concurrently while maintaining the accuracy of the results. In some methods, the accuracy of the method is influenced by the number of alternative services or the increase in the number of criteria. Therefore, by increasing the number of criteria or properties in the selection mechanism, the accuracy should not be affected. In fact, scalability depends on the accuracy of the approaches. Therefore, scalability cannot be evaluated if the accuracy of the approach is not reliable.

3.2 Comparison at the First Level

As discussed in Section 2.5, the proposed classification is similar to a tree that includes two levels, first level and second level. Also the second level includes two levels, semantic level and MCDM level. Those levels are compared individually. The comparisons are done based on some criteria that described in prior sections. Those criteria are derived based on the most common criteria in the existing researches related to this research.

The rates and data of each table are provided based on the approaches described in literature. If an approach addresses all aspect of a criterion the given rate is "*High*". If only a part of the expected aspects is fulfilled the rate would be "*Average*". Finally in the situation that none of the aspect of a criterion is addressed "*Low*" rate is assign for that approach and criterion.

The value of each cell is explained below for each related approach. The overall result follows from the formula below:

$$x_i = the\ value\ of\ each\ cell\ that\ can\ be\ evaluated\ by\ the\ following$$

Low = 1 Average = 2 High = 3

$$\text{Overall Mark (OM)} = \frac{\sum_{i=1}^{n} x_i}{n}$$

$$\text{Overall Result} = \begin{cases} Weak & if\ \ OM \leq 1.5 \\ Average & if\ \ 1.5 < OM < 2 \\ Good & if\ \ OM \geq 2 \end{cases}$$

The comparison of the first level based on this formula is shown in Table 3.1.

Table 3.1 Comparative evaluation of WSS at the first level

		Criteria					Overall Result
Approaches	Instance	User Preference	Performance	Accuracy	Automatic	Scalability	
Improve Protocol	[53]	Average	Low	Average	Average	Low	Average
Semantic	[4]	Average	Average	Average	Average	High	Good
	[62]	Average	Low	Low	Average	Average	
Policy	[46]	Average	Low	Low	Low	Low	Weak
	[9]	Average	Low	Average	Low	Low	
Trust & Reputation	[48]	Low	Average	Low	Low	High	Average
	[14]	Average	Average	Low	Low	High	
MCDM	[67]	Average	Average	Average	Average	Average	Good
	[52]	Average	High	Average	Low	Average	
	[8]	Average	High	Average	Low	High	
	[65]	Low	High	Average	Average	High	
Atypical DM	[71]	Average	Low	Low	Low	Average	Weak

Improve Protocol: This approach considers user preference. Performance is not adequate because service data and information quality are separated, which either causes the registered services to be monitored dynamically by the quality broker or causes service details to be stored in more than one location by the SP. Furthermore, the automation of this approach is only average because it is not a fully automated approach. The performance decreases when the number of properties is increased. Therefore, this approach cannot be considered scalable.

Semantic: This approach uses user preference in the selection process but it is not sufficient. The performance of these approaches is unsatisfactory because the overhead of the selection algorithm is high. The approach is semi-automated because the selection mechanism in this approach is not fully automated and the human ultimately selects the service. The accuracy of this approach is good when there are a large number of properties. Therefore, this approach can be considered scalable.

Policy: One of the worst approaches in WSS is the policy-based selection approach. Other than user preference, the remaining selection criteria are unsatisfactory. This approach focuses on user preference but is poorly defined as a selection mechanism, as mentioned in 2.5.1. Thus, the performance is poor. The human involvement is a disadvantage of this approach. Finally, the policy-based approach does not allow for a large number of properties, so it cannot be considered scalable.

Trust and reputation: Although trust and reputation relies on trust and taking the feedback of users, the user preference is low. Performance cannot be high because approaches do not consider the evaluation functions for different criteria. In most of the approaches based on trust and reputation, service selection is performed manually during design. Thus, automation is not essential. The accuracy of the approach remains high when there are a large number of properties because this approach is based on trust and does not emphasise the properties as much as other approaches. Therefore, the approach is scalable.

MCDM: The non-functional properties are fundamental for MCDM approaches and point to user preferences. Thus, the impact of applying user preferences is high. The performance of the approach is adequate because the performance of the mathematic algorithms is proven, but the automation is low because the requester selects the service manually in the suggestion service list. The scalability is sufficient because most of the algorithms that are applied are scalable.

3.3 Comparison at the Second Level

Because this research is related to semantic and MCDM approaches, the second level of those two approaches is examined. The comparison of the second level is divided into two tables with specific criteria.

3.3.1 Semantic Level

The semantic levels of different approaches are compared in Table 3.2.

Table 3.2 Comparison at the semantic level

		Criteria					Overall Result
Approaches	**Instance**	**User Preference**	**Performance**	**Accuracy**	**Automatic**	**Scalability**	
WSMO	[4]	Average	Average	Average	Average	High	Good
OWL-S	[62]	Average	Low	Low	Average	Average	Average
SAWSDL	[64]	Average	Average	Low	Average	Average	Average

The conclusion below is based on the discussion in Section 2.5.4.

WSMO: The algorithm in Section 2.5.4.1 includes user preference, but it has an overhead that affects performance. This approach can only be semi-automated because the human finally selects the service among the top alternatives of the service list. Furthermore, the approach can be used for a large number of properties without decreasing the performance because the algorithm already addresses multi-criteria.

OWL-S: This approach considers user preference because the recommendation system is based on user feedback. The performance will decrease by using the graph and recommendation system. Similar to the previous approach, this approach is not fully automated. The scalability of the approach is average because the accuracy is affected when the number of properties increases.

SAWSDL: Because the matching mechanism prepares more detailed feedback, the user preference is more fully considered. In [13], the average performance is proven. Automation is average due to human involvement, and the scalability is also average because the performance is affected by an increased number of properties.

3.3.2 MCDM Level

This section summarises the approaches discussed in Section 2.5.5. The criteria at this level differ from those of prior levels because this stage is more algorithmic, whereas the approaches described in prior levels are concerned more with modelling. However, the maturity of criteria is the same as before; the criteria include QoS, user preferences, accuracy, confidence level (CL), automatic and scalability. The definitions of the additional two criteria are as follows:

- ✓ QoS: QoS refers to the approaches that consider QoS as the criterion for decision making. The prevalent QoS are duration, reliability, availability and cost. QoS is a subject of utmost importance, and thus, this criterion must be considered.

- ✓ CL: is the confidence level of the SP. This criterion refers to how the approaches use the CL of the SP, namely, whether they consider the accuracy of the data or trust SP.

Table 3.3 Comparison at the MCDM level

	QoS	User Preference	Accuracy	CL	Automatic	Scalability	Overall Result
[71]	✗	Average	Low	✗	Low	Average	Weak
[65]	✓	Low	Average	✗	Average	High	Good
[66]	✓	Average	Low	✗	Low	Low	Weak
[67]	✓	Average	Average	✗	Average	Average	Good
[68]	✓	Average	Average	✗	Low	Low	Average
[8]	✓	Average	Average	✗	Low	High	Good
[69]	✓	Low	Average	✗	Low	Low	Weak
[70]	✓	Low	High	✗	Low	Low	Average
[52]	✗	Average	Average	✗	Low	Average	Weak

An overview of the methods is provided in Table 3.4.

Table 3.4 Summary of the MCDM approaches

Methods	*Fuzzy*	*ANP*	*Linear Programming*	*TOPSIS*	*AHP*	*PROMETHEE*	*Naive DM*
[75]	✓						
[65]	✓		✓				
[66]							✓
[67]				✓			
[68]	✓			✓			
[8]	✓			✓			
[69]		✓					
[70]						✓	
[52]					✓		

3.4 Summary

In this chapter, the existing WSS approaches have been investigated and evaluated based on the defined criteria. A cell representing the overall result for each approach is provided at the end of each row. This cell is prepared using the formula in Section 3.2. The result of each table for every category is discussed separately.

The study demonstrates that several approaches have the capability to support WSS in terms of modelling, the algorithm and decision making. The above evaluation indicates that none of the current approaches support the CL of the SP. Furthermore, the ability to verify the accuracy of input data are lacking. Although the accuracies of most of the approaches are average, no approach supports both aspects of accuracy, i.e., the accuracy of the data and algorithm. However, designing an accurate approach to support WSS should consider both types of accuracy. It is essential that the approach uses the CL of the SP to support the accuracy of the input data and that it uses precision and recall for the result of the approach.

As described in Section 2.5, there are three aspects for WSS, namely, *modelling QoS*, *data gathering of QoS* and *algorithm*. The overall result demonstrates that the semantic-based approaches can provide a suitable environment for modelling WSS. Among these approaches, WSMO addresses most of the criteria. As shown in Table 3.1, the overall result of trust and reputation is better than that for policy-based approaches. Moreover, MCDM approaches are shown to be preferable. Therefore, this research focuses on WSMO as the modelling aspect, trust and reputation as the data gathering aspect and MCDM as the algorithmic aspect. Based on the selected aspects, this research proposes an accurate and flexible approach that addresses the current issues of service selection.

CHAPTER 4

RESEARCH METHODOLOGY

This chapter describes the methodology of this research, including the research design, research procedure, operational framework, instrumentation, assumptions and limitations.

4.1 Research Design

The research problem of software engineering is classified by several research projects [76-79]. There are four methods, namely, *scientific, engineering, empirical* and *analytical*. However, only two methods, *scientific* and *engineering,* are common.

A *scientific* method develops a theory to describe a phenomenon. The way to prove this theory is by providing a hypothesis and testing alternative variations of the hypothesis and data collected to verify or refute the claims of the proposed hypothesis [79]. A Scientific problem concerns the study of an existing object such as algorithmic complexity, software metrics or testing techniques [76].

An *engineering* method is concerned with a solution that develops and tests a hypothesis [79]. Engineering problems involve the construction of new objects, such as models, approaches, methodologies and techniques [76]. In an *engineering* method, it is essential to study existing methodologies, evaluate them to verify their disadvantages and advantages and then propose a new approach that resolves the existing disadvantages while retaining the advantages of the existing investigated approaches.

This research develops a new selection approach to support semantic web services. However, as described above, producing new approaches is an engineering problem. Therefore, this research is directed at engineering design in terms of modelling, constructing and evaluating the new object. The flowchart of research design is provided in Figure 4.1.

4.2 Research Procedure

The research procedure, which includes and briefly describes each step of this research, is shown in Figure 4.6. The steps of the research procedure are *literature review, analysis of requirements, development* and *evaluation.* These steps are explained in more detail below.

4.2.1 Literature Review

In this phase, there are three stages of reviewing the literature. First, the definitions of preliminary concepts of semantic web services are identified. Second, the current methods and technologies of semantic web services and MCDM, which can be found in Sections 2.3 and 2.5.4, are investigated. Finally, the WSS approaches that are most relevant to this research are discussed. This stage includes an investigation of the current approaches in terms of the modelling and algorithm aspects to identify the processes of various WSS approaches. The results of this stage are classified and presented in Section 2.5.

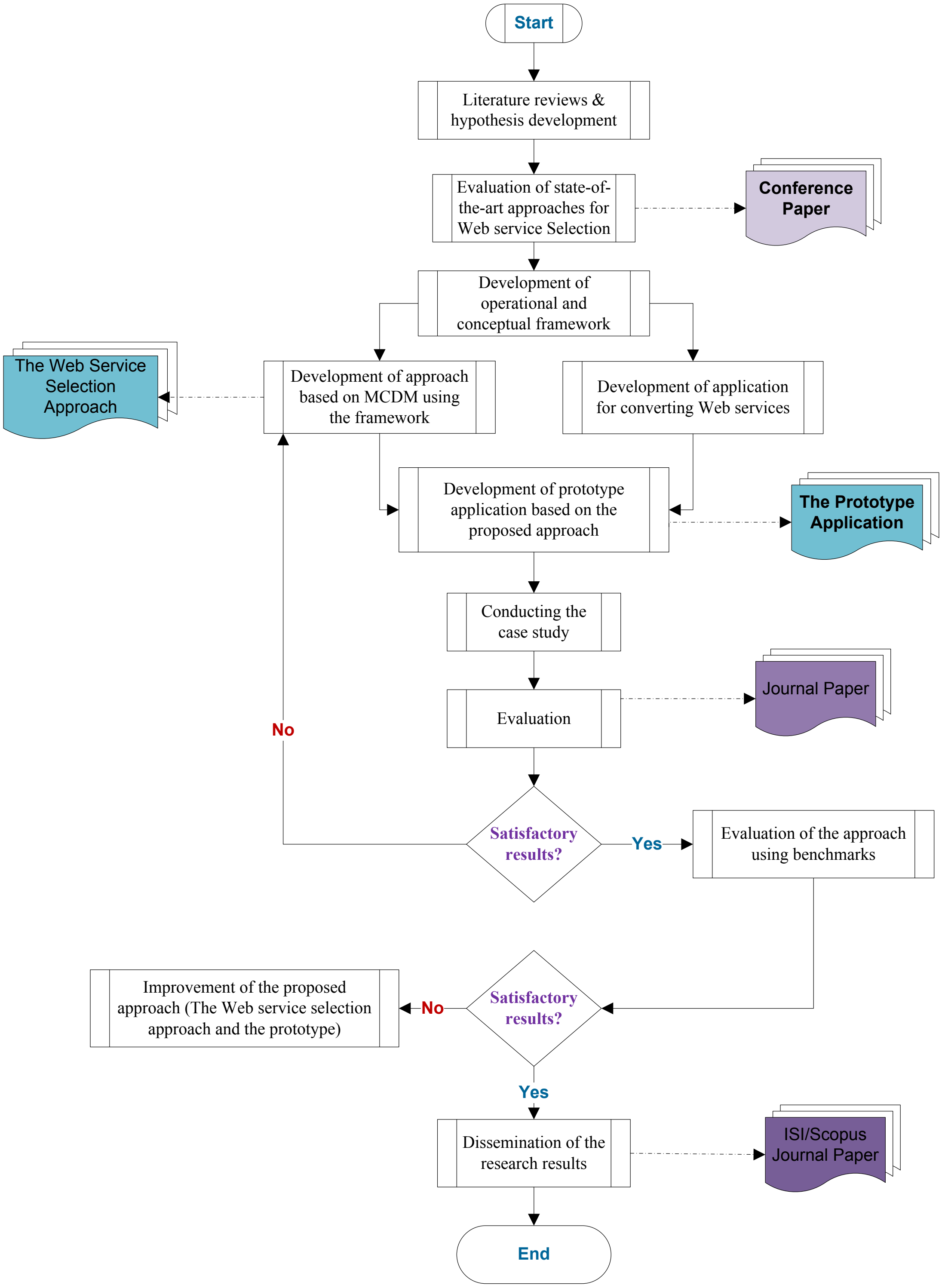

Figure 4.1 Research Design

4.2.2 Analysis of the Requirements

This phase has two major stages. First, the approaches discussed in Section 2.5 are evaluated based on the common criteria provided in Section 3.1. The outcomes of this stage are three comparative evaluation tables. The rates and data of each table are provided based on the approaches described in literature. If an approach addresses all aspect of a criterion the given rate is "*High*". If only a part of the expected aspects is fulfilled the rate would be "*Average*". Finally in the situation that none of the aspect of a criterion is addressed "*Low*" rate is assign for that approach and criterion. The result of this evaluation is discussed, and the advantages and disadvantages of each approach are clarified.

Second, based on the results of the evaluation, the approaches used in this research for the three aspects of service selection are specified. These approaches are WSMO, MCDM and trust and reputation. The figure below presents the proposed classification of the current approaches and the approaches that are suitable for this research (red text).

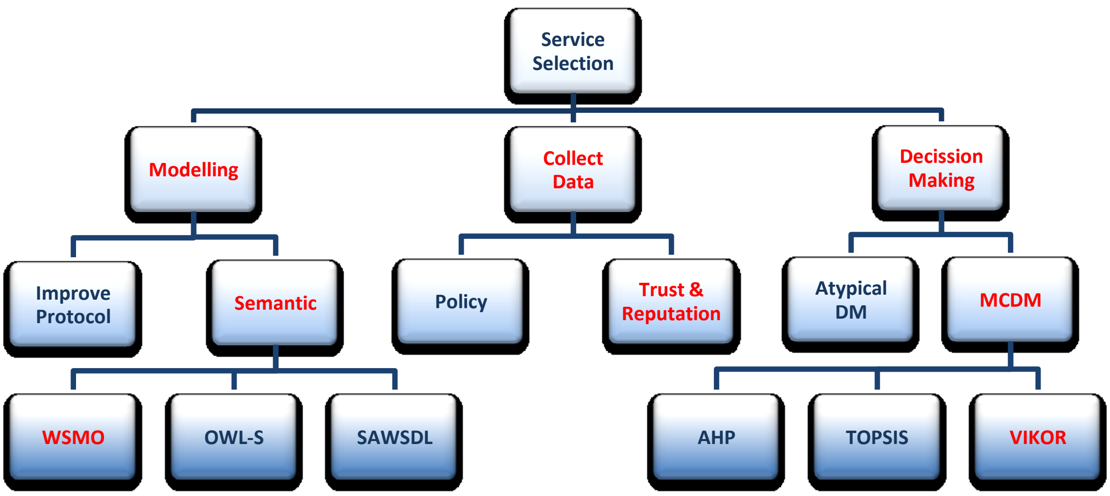

Figure 4.2 Direction of research based on the evaluation

4.2.3 Development

In the development phase, a new approach is proposed based on the output of the prior phase to eliminate the limitations of the current approaches to achieve the research objective. The proposed approached, termed AMW, is described in Section 5.2. This approach consists of *architectural* and *algorithmic* aspects. The architecture of AMW was described previously, and six algorithms are proposed below to support the former aspect. These two aspects are detailed in Sections 5.3 and 5.4. In addition, a prototype to support the applicability of AMW is developed. Finally, the result of this stage is presented in Section 6.4.

4.2.4 Evaluation

The applied evaluation method of this research is *quantitative* and *qualitative*. The aim of the evaluation of the proposed approach is to systematically confirm that AMW meets the objective of this study by supporting *accuracy* measurement. Accuracy is a quantitative metric that is typically measured by precision and recall [73]. Also, the F-measure, which was proposed by Rijsbergen [80], is applied as the harmonic mean of precision and recall. These measures usually are used for the unranked retrieval results, on the other hand this research is based on ranking results; therefore for the ranked retrieval results, it is needed to extend these measures or define new measure. In the accuracy measurement section the way of defining measure for ranked retrieval is discussed.

Regarding the qualitative criteria of this research, flexibility and automation, the DESMET method proposed by Barbara Kitchenham [81] is employed. DESMET is a methodology used to evaluate software engineering methods and tools. There are nine manners proposed by DESMET [82], including quantitative and qualitative manners. Feature analysis is selected as the purpose of the qualitative method in this research. The evaluation steps of each manner are described below.

4.2.4.1 Accuracy Measurement

The metrics of accuracy can be evaluated using two case studies and the experimental results of the developed prototype. The results of each stage are presented separately. Achieving the comprehensive and overall results can help to verify the accuracy of the proposed approach.

Two case studies are applied: *flight booking* and *hotel reservation*. In both case studies, six scenarios are employed to cover all states that might occur. Any possible state would be represented by one of the six described scenarios. The results of each scenario are presented via precision-recall graphs. The results of this stage are presented in Section 6.3.

To evaluate the proposed approach, the experiment is performed on the first version of AMW test collection (AMW_TC1). The OWL-S test collection (accessible via http://projects.semwebcentral.org/frs/?group_id=89&release_id=380) is converted into a WSML file, and then the non-functional properties of each service are added to the services. Among them the category of travelling which includes 197 web services is chosen for this research. Furthermore, the evaluation based on the experiment is performed on a computer with the Windows Vista Ultimate Edition operating system, an Intel Core 2 Duo 3.00 GHz processor and 2 GB of RAM using JDK 1.6.0. The results of this stage are presented in Section 6.4.

As mentioned, precision and recall are the standard measures of accuracy. The general concepts of these measures are shown in Figure 4.3 [73]. In following the general concepts of precision and recall are described:

$$\text{Precision (P)} = \frac{tp}{tp+fp} \quad \text{(a)}$$

$$\text{Recall (R)} = \frac{tp}{tp+fn} \quad \text{(b)}$$

tp: true positives; ***fp***: false positives; ***fn***: false negatives

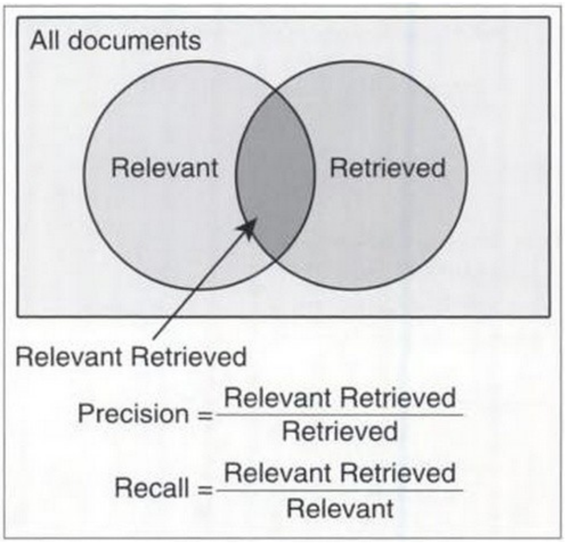

Figure 4.3 General concepts of precision and recall

However, these measures should be extended for the ranked retrieval results. In a ranked retrieval context, the appropriate sets of retrieved documents are naturally given by the top k retrieved documents. For each such set, the precision and recall values can be plotted to give a precision-recall curve, such as the one shown below [83]:

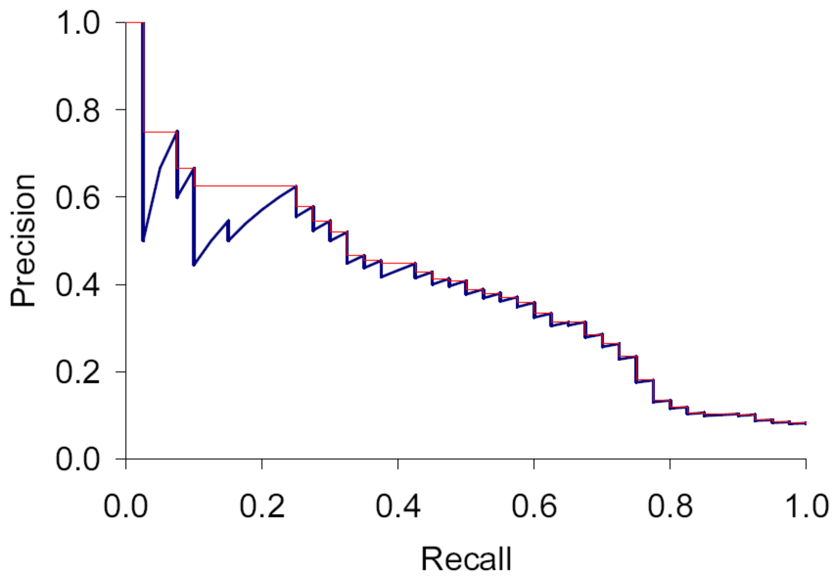

Figure 4.4 precision-recall graph

As observed in the above figure, precision-recall curves have a distinctive saw-tooth shape. It is useful to remove these teeth, and the standard method is to use an interpolated precision: the interpolated precision at a certain recall level r is defined as the highest precision found for any recall level r′ ≥ r [83]:

$$p_{interp}(r) = \max_{r' \geq r} p(r')$$

The interpolated precision is shown by a thinner (red) line in Figure 4.4. Examining the entire precision-recall curve is very informative, but it is often desirable to summarise this information with a few numbers, or perhaps even a single number. The traditional method is the 11-point interpolated average precision. For each ranking, the interpolated precision is measured at the 11 recall levels of 0.0, 0.1, 0.2, . . ., 1.0 [83]. The interpolated precision for the previous graph can be observed below:

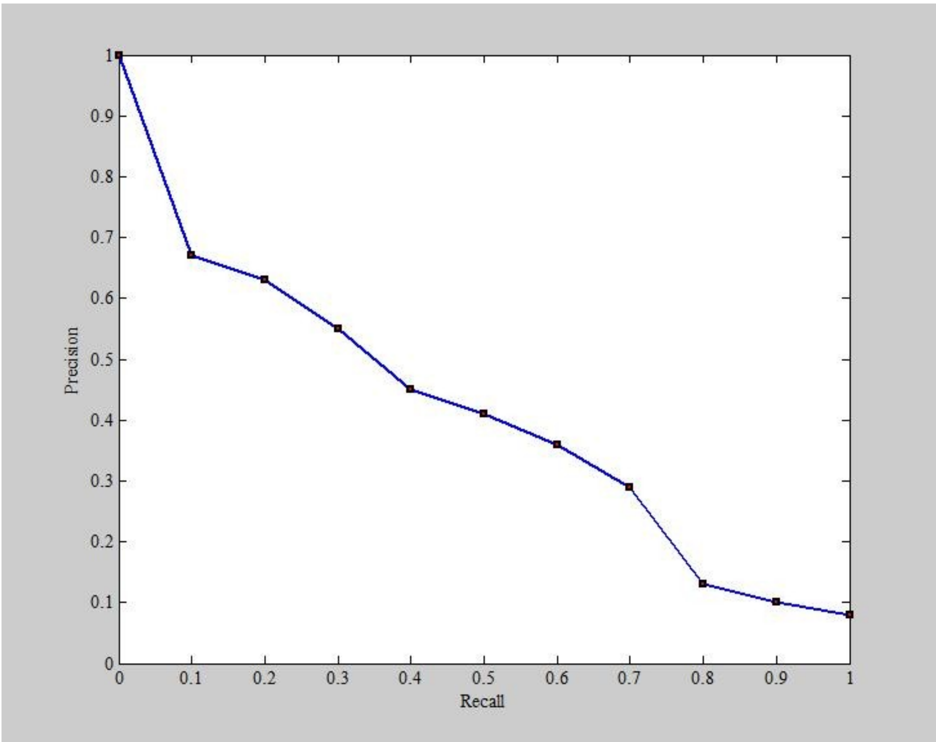

Figure 4.5 Sample of averaged 11-point precision-recall graph

Curves closest to the upper right-hand corner of the graph (where recall and precision are maximized) indicate the best accuracy. Comparisons are best made in three different recall ranges: 0 to 0.2, 0.2 to 0.8, and 0.8 to 1. These ranges characterize high precision, middle recall, and high recall, respectively [84].

4.2.4.2 Feature Analysis

As mentioned earlier, the type of analysis for qualitative metrics is *feature analysis,* as suggested by Barbara Kitchenham [81]. DESMET proposes three steps for feature analysis: *identifying features*, *scoring features* and *analysis*. These steps are described below.

(i) **Identifying Features:** The features that need to be assessed are flexibility and automation. Flexibility refers to the ability of the approach to work in different situations; if the circumstances are changed, a flexible approach can adapt itself to the new condition. Automation refers to the ability of the approach to perform the process of selection without interaction by the service requester. These features also consist of sub-features, such as Extra QoS, Expert DMs, Default Weights and Goal Generator.

(ii) **Scoring Features:** Each feature should be accompanied by an assessment of its *importance* and *conformance.* The scales for measuring importance and conformance are discussed below.

- ***Importance:*** The importance of a feature can be assessed by considering whether it is mandatory or only desirable. This view of importance leads to two assessment criteria; one identifies whether a feature is mandatory, and the other assesses the extent to which a non-mandatory feature is desired. To assess a feature, the following scale points are considered: mandatory, highly desirable, desirable and nice to have.

- ***Conformance:*** The aims of the assessment scale for conformance are defining what level of support is required for a particular feature and providing the assessor with a consistent measurement scale against which to score the feature of a particular candidate.

In Chapter 6, the values of importance and conformance of each feature are identified.

(iii) **Analysis:** After providing the importance and conformance of features, the score sheets must be analysed, and the approach that achieves the best score must be determined. Based on the DESMET method, if the acceptance threshold is explicated, the analysis should be based on the difference between the acceptance threshold for each feature set by the users and the score that each approach obtained for the feature. If the acceptance threshold is not achievable, the assessment should be based on the scores of the approaches relative to one another. As the acceptance threshold is not achievable in this research, the latter approach is used. Therefore, the analysis must be based on accumulating the absolute scores. The combined score for one feature set would be the sum of the conformance values of all features for a certain approach, which can also be represented as a percentage of the maximum score. For example, suppose that the combined flexibility score with three sub-features is 10 out of 15 (the maximum score). The converted percentage score would be 67%. Finally, the overall score can be obtained by determining the aggregate score for each feature set, as described in Chapter 6. The result is presented in Section 6.5.3. The results can also be represented by "multiple-metric graphs" [85], which are a variation of Kiviat diagrams. In such graphs, the size of the slice represents the importance of the feature relative to the other features displayed. Therefore, a larger slice represents a more important feature.

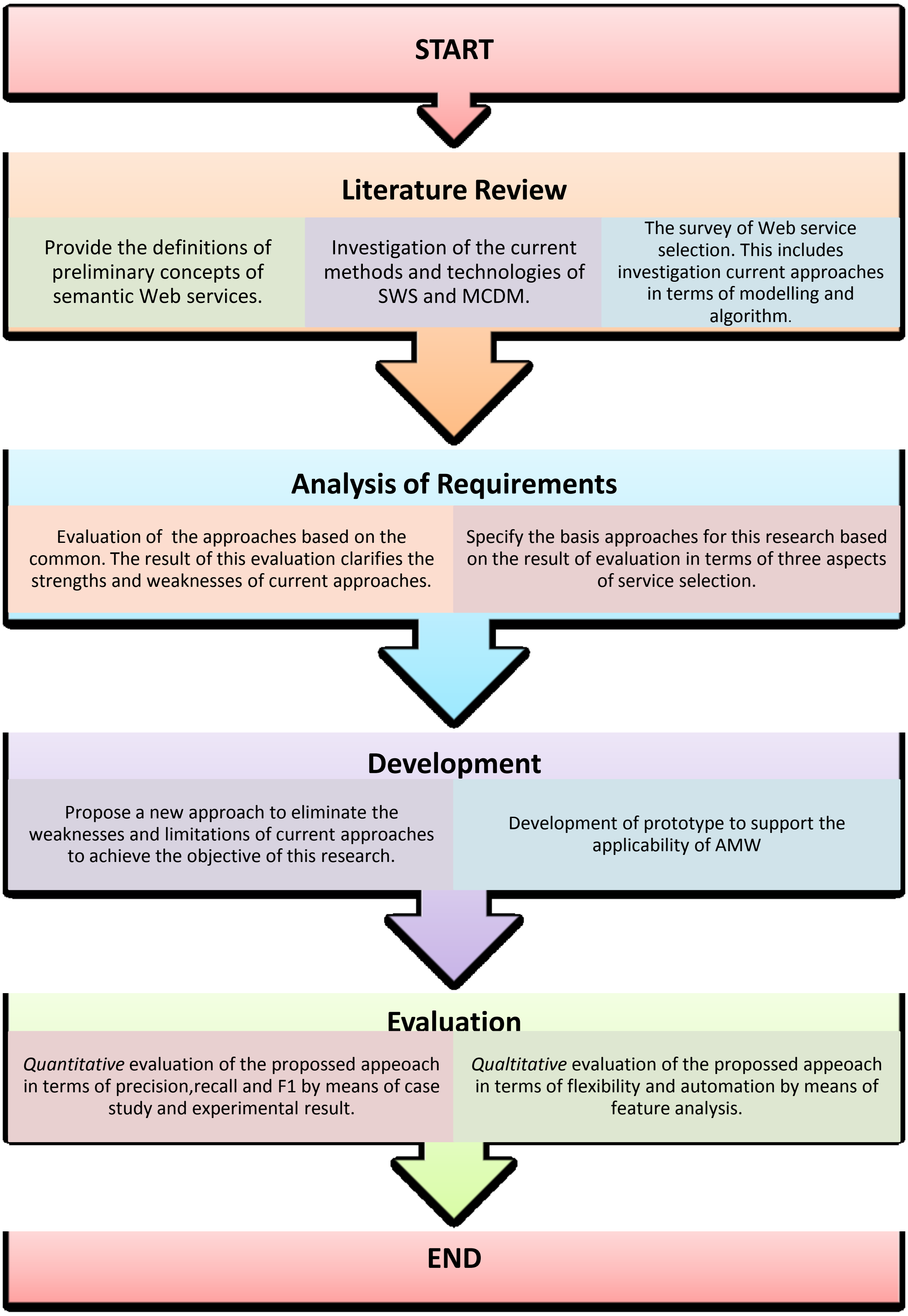

Figure 4.6 Research Procedure

4.3 Operational Framework

In this section, the research operational framework that is applied in this research is provided. The operational framework provides the relations between the research questions and research objectives. The research objectives are developed based on the research questions. Required activities that support each objective are provided, and the deliverables for each objective are specified. The results of this section as an operational framework of this research are provided in Table 4.1.

4.4 Instrumentation

The WSMX, which is the excitation environment of WSMO, is developed and programmed in Java. Thus, for future integration and extension, the software prototype is also developed in Java. Because the prototype will be tested on the web or other environments for simulation, web environments, such as Apache or Apache Tomcat, are used.

In addition, to design the proposed approach, such tools as *WSMO studio, WSMX*, *WSML Editor* and *OWL2WSML* will be provided. *WSMO* Studio is used to design a web service in a WSMO model. Users can apply ontologies, describe the Goal and use the mediator. Moreover, *WSMX* is employed to convert the OWL file to the format of the WSML file.

Moreover, *MATLAB* is used to plot the results; for this purpose, one function is applied on the results of the experimental and case studies. This function is based on the concept of the precision-recall graph for ranked retrieval results described in Section 4.2.4.1.

Table 4.1 Operational Framework

	Research Question	Objective	Activity	Deliverable(s)
1	Why are the existing approaches unable to answer the present problem in WSS?	To investigate and evaluate the state of the art in WSS approaches.	-Literature study - Comparative evaluation of current approaches	- Literature review
2	How can accurate data be prepared, using extra QoS, and achieve the best results, particularly in terms of the accuracy of supporting WSS?	To develop a new approach for WSS in which the accuracy of both the input data and selection method are considered.	- Building a model for WSS - Designing the prototype tool - Integrating the tool	- The WSS model - Design documentation - Source code - Executable tool
3	What are the main elements of the accurate approach for selecting web services?	To propose default criteria weights using the AHP method to help users express their preferences.		
4	How to select best service via the accurate approach?	To design and formulate algorithms to support WSS.	Designing algorithms to improve existing WSS approaches	- Source code of the algorithm
5	How to validate and evaluate the effectiveness of the proposed approach to support WSS?	To evaluate the accuracy capability of the proposed approach by the developing a selector tool and comparing it with other approaches.	- Measuring the precision, recall metrics - Analysing the precision, recall values - Applying the proposed approach on the case study	- Precision, recall values - Analyse results in term of precision, recall values

4.5 Assumptions and Limitations

In this research, the following assumptions and limitations must be considered:

(i) This research does not provide the reasoner solution. It is assumed that the current reasoner in the WSMO environment is efficient and able to perform the matchmaking based on the semantic descriptions. The approach is merely concerned with the selection of discovered services based on the requested goal.

(ii) It is assumed that the provided data by expert DM, which are used for the MCDM service selection, are correct and that the expert DMs are able to express the rates of alternatives.

(iii) The research assumes that the goal file is formatted in WSML. Goal files that are in the other formats should be converted to the WSML file format first. As the OWL file format is a format for which there are several repositories, there is a reused component in the approach that converts OWL files to WSML files.

4.6 Summary

In this chapter, the research methodology that is adopted in this research is described. In the first section, the research design is discussed to identify the required stages. The second section describes the research procedures in detail and consists of four sections: literature review, analysis of requirements, development and evaluation. The following sections discuss the operational framework and research instruments. Finally, the assumptions and limitations of this research are described. In the next chapter, the proposed approach and development of the software prototype are discussed.

CHAPTER 5

THE AMW APPROACH TO SUPPORT WSS

In this chapter, the proposed approach, called AMW, is described. This approach fixes the issues discussed in Chapter 3. The proposed solution for providing flexible and automated service selection involves the application of a flexible framework and proficient MCDM method. The AMW approach involves a framework as an architectural aspect and a formula as an algorithmic aspect. The proposed approach considered the accuracy of both the input data and selection method. First, the limitations of the existing approaches are reviewed, as described in Section 2.5. Next, the AMW approach, including its architecture and algorithms, is described in detail. Furthermore, the proposed convertor, which facilitates the automation of AMW, is described. This stage automatically converts the needs and preferences of users to WSML. Finally, the implemented prototype for the proposed approach is described in detail.

5.1 Limitations and Restrictions

The existing WSS approaches have been discussed in detail in Section 2.5. In Chapter 3, those approaches are compared based on specific criteria. After the evaluation stage, the advantages and disadvantages of those approaches are clarified. In this section, those approaches are reviewed, and the basis of this research and the reasons for which those approaches are chosen are discussed.

As described in Section 2.5, three aspects should be considered in any WSS approach: modelling, data collection and decision making. As discussed in Section and based on the results shown in Table 3.1 and Table 3.2, the appropriate approaches as a basis for this research are WSMO (2.5.4.1) and trust and reputation (2.5.2) for modelling and data collection, respectively.

In addition, the VIKOR method (0) is employed to improve features, which is absent in the literature for service selection but is well known in other research. VIKOR is selected as an MCDM method in this research.

The limitations and restrictions of existing approaches, which this research attempts to resolve, are highlighted below.

- ✓ In the current approaches, the criteria used in the selection algorithm are limited to non-functional properties and QoS, such as reliability, availability, performance and cost. However, there is an important factor in the discovery stage that is very useful for selection. This factor, called the *type of matching*, is not considered in the existing approaches as an important criterion for selecting the appropriate service.

- ✓ Data collection is the basis of service selection because accurate results cannot be obtained with incorrect data, even if the algorithm is efficient and accurate. Data collection is performed in some approaches by trusting the SP. The SPs advertise their services as they want. In this circumstance, the SP is trusted with no consideration of the reliability and CL of SPs.

- ✓ In some approaches, the data can be collected from the expert DM, who has sufficient knowledge about the system and can offer the data needed for the service selection. In those approaches, the power of the DMs is lacking, particularly in the situation of group decision making. In group decision making, the data are collect from more than one expert DM. Some of the DMs might have more experience than others, but there is no indication of how experience level is considered.

- ✓ As discussed in Section 3.1.4, the automation of service selection is one of the challenges in this research area. In the existing approach, the goal WSML file is generated manually, meaning that the preferences of a user are added to the goal WSML file by that user.

The proposed approach attempts to cover the above issues as follows:

- ✓ Consider *type of matching* as a criterion in the service selection algorithm.
- ✓ Employ the *CL* of the SP to examine the reliability of the SPs.
- ✓ Apply the *power of the DMs* to prove the levels of DMs and determine the accuracy of their knowledge.
- ✓ Use of a *convertor* to convert and translate the user preferences into the goal WSML format.

The next section describes the proposed approach and explains how the approach solves the current issues.

5.2 The AMW Approach

This section discusses the proposed approach, called AMW (an Accurate approach based on MCDM and WSMO. This approach is divided into *architectural* and *algorithmic* aspects. In the architectural aspect, framework and architecture are described, and in the algorithmic aspect, formulas and algorithms that support the model and framework are described.

5.3 The Architectural Aspect of AMW

In this section, the general architecture of the AMW approach is described. Then, the mechanisms of service selection in abstract and concrete views are discussed. The architecture of AMW is presented in Figure 5.1.

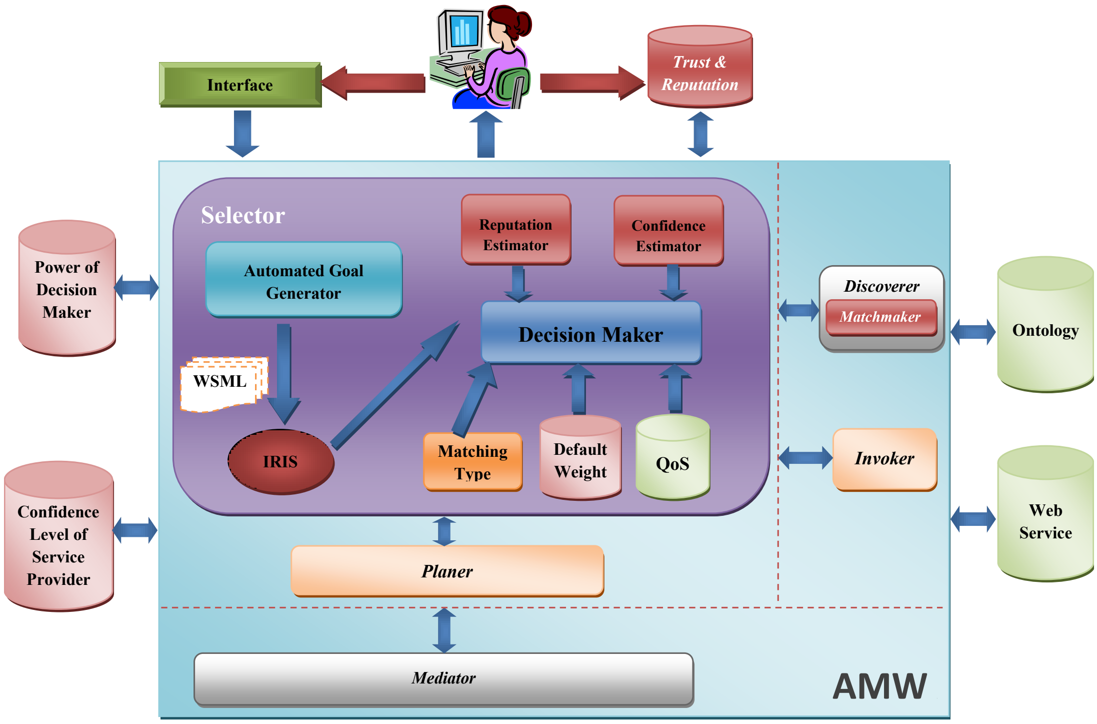

Figure 5.1 General architecture of the AMW approach

The overall process is involved three stages: pre-selection, selection and post-selection. The pre-selection stage concerns what should be prepared for the selection stage. For the selection service, the QoS weights and the rates of alternatives must be prepared. The involved components are: Power of DMs, CL of SP, QoS, Default Weight and matching type. In the selection stage, based on the data, which are prepared in the previous stage, decision making is performed. Finally the post-selection stage updates the databases of CL of SP and trust and reputation. The involved components in this stage are reputation estimator and confidence estimator.

As discussed in Section 5.2, the proposed approach is based on WSMO. Thus, the approach has some of the same components as WSMO, such as *mediator*, *planner, discoverer, ontology* and *web services.* Therefore, as those components are described in detail in [4, 19, 86-88], this research only provides a brief description.

- ✓ **Discovery:** Once the user semantically specifies the needs with the goal, WSMX can execute two types of discovery to find matching services: *keyword-based discovery* and *semantic-based discovery*. In performing discovery, the web service and goal can have various types of matching:

  - ❖ *Exact match:* the goal requirements are provided by WS exactly.
  - ❖ *Subsumption match:* a part of the goal requirements is provided by WS.
  - ❖ *Plug-in match:* the goal requirements and additional functionality not required by goal are covered by WS.
  - ❖ *Intersection match:* a part of the goal requirements and other worthless functionality are provided by WS.
  - ❖ *Non-Match:* none of the goal requirements can be covered by WS.

- ✓ **Mediator:** The selection components and discovery consider the semantic descriptions of the web services and goals, and based on these descriptions, some operations must be performed. If different ontologies are used to define these elements, these two components must perform their functionalities by the services of data mediators. Depending on the type of elements that the mediators bridge or link, the mediators are divided into four types: 1) web service-to-web service mediators, 2) web service-to-goal mediators, 3) goal-to-goal mediators and 4) ontology-to-ontology mediators [4].

- ✓ **Planer:** This component is used to plan for candidate web services to be composed together to fulfil the user's goal. This component is also termed the web service composer.

- ✓ **Selector:** This component is the principal component in this research that uses other parts, such as the ranking system, quality ranking and matching ranking, to find the service or services that satisfy the user requirements and QoS. The mechanism of selection is described in Section 5.3.3.

5.3.1 AMW Components

The components added by AMW are: *CL of SP, power of the DMs, matching type, default weights, reputation estimator, confidence estimator, interface, goal generator* and *DM.* These components are described below.

5.3.1.1 Interface

First, the user should express her preferences in the Goal WSML file, which is traditionally performed by the user. This step takes time and requires knowledge and familiarity with the WSML language. In the AMW approach, one interface is provided between the user and system. The interface can take user preferences and send them to the goal generator.

5.3.1.2 Goal generator

The data from the interface are the input of the goal generator. This component converts user preferences to the WSML format automatically. The goal generator must improve the automation of AMW. In the proposed approach, there are three types of non-functional properties: *QoS, type of matching* and *reputation*, as shown in the following table:

Table 5.1 Mapping the goal generator table

Type of NFP	How data are generated	User involved
QoS	Expressed by user	Yes/No
Type of matching	Originates from the discovery component	No
Reputation	Generated by AMW	No

As shown in Table 5.1, the only NFPs that may involve the user is QoS. However, AMW offers the default criteria weights that are discussed in Section 5.3.3, which demonstrates that the AMW automation is adequate. The algorithm of goal generator is described in Section 5.5.3.

5.3.1.3 CL Estimator

One of the processes that should be performed after the discovery is the updating of the CL of SPs (Section 5.3.3). This component estimates the CLs of the SPs and then maintains a bank of the CL of SP updated. The feedback and score given to SP is added to the current CL score and the new CL of SP is updated to the assigned bank.

5.3.1.4 Reputation Estimator

Similar to the CL estimator, this component takes the feedback of users and then evaluates that data. Finally, the estimated reputation of the services will be sent to the trust and reputation bank, and the bank of the power of the DM is also updated. The feedback of the user is added to the current reputation score of the selected service, and the latest reputation score is updated in the trust and reputation bank. The algorithm of these factors is described in Section 5.4.3.2.

5.3.1.5 Default Weight Estimator

This component estimates the default QoS weights. For this purpose, the opinions of expert researchers are collected as input for the AHP method. After employing AHP, the default weights are provided and the bank of the default QoS weights is updated accordingly.

5.3.1.6 Decision Maker

The decision maker is a central and important part of the selector. All of the described components lead to this stage, in which the result is provided and the services are ranked. Next, the service at the top of list is selected for execution. As this approach is dynamic, the second service in the list automatically replaces the first if some error occurs at runtime.

The core of the decision-maker component is an improved VIKOR. The VIKOR method is described in Section 2.3.3, but in the AMW approach, the VIKOR features are enhanced. VIKOR has two shortcomings:

1) If the rates of all alternative services have the same value for a certain criterion, for example, all discovered services have the same price, VIKOR is not able to resolve this issue. AMW enhances the VIKOR so that it is able to handle this problem.

2) In some cases, one service has the highest value for most criteria but does not address some of the criteria. For example, one service has the highest value for performance, reliability and response time, but the rate of availability of the service is close to zero. In this case, VIKOR again provides the wrong result. Because it is based on mathematics, VIKOR gives the highest rank to a certain service that cannot support

one criterion. However, AMW detects and prevents this issue. The VIKOR algorithm with improved features is described in Section 5.4.2.

5.3.2 Abstract View of the Service Selection Mechanism

After discovery, a list of services that can satisfy user requirements will be available. C1) Occasionally, a single web service can respond to the user's requirements. C2) Occasionally, multiple web services are discovered. C3) In contrast with previous situation, there is no single service to support user requirements after discovery; thus, a set of single discovered services should be composed. However, this stage is broken down into: C3_1) Only one web service is discovered for each functionality and C3_2) more than one web service is discovered. The aim of this research is to address the selection process in situations C2 and C3_2.

At this time, each discovered service has a type of matching (*exact, plug in, subsumption, intersection and non-match*). These types of matching can be classified as follows:

- ✓ ***Non-Match:*** In this situation, the process will be fetched, and there is no selection and composition.

- ✓ ***Exact Match:*** This situation can have two meanings: either only one service is discovered as an exact match or more than one service is an exact match. In the first situation, there is no selection process because there is no option to use the selection method and only one service is available. However, in the second situation, there are some available services that are exact matches. Thus, the selector should select the best service based on the aim of this research (C2).

- ✓ ***Other Matching:*** In this state, one service cannot satisfy the user's requirements; thus, a set of discovered services should be composed to develop a service to satisfy the user's needs. In this situation, if multiple services are discovered, the selector chooses the best service based on the user requirements (C3_2).

In addition, the type of matching is used as an element of selection that can play an important role because it can affect QoS factors, such as performance and cost. Therefore, each factor can have its own rating between zero and one, where one is used for an exact match and zero for anon-match. The details are described in Section 5.4.1.2. These descriptions are illustrated below as an abstract flow chart of the proposed framework.

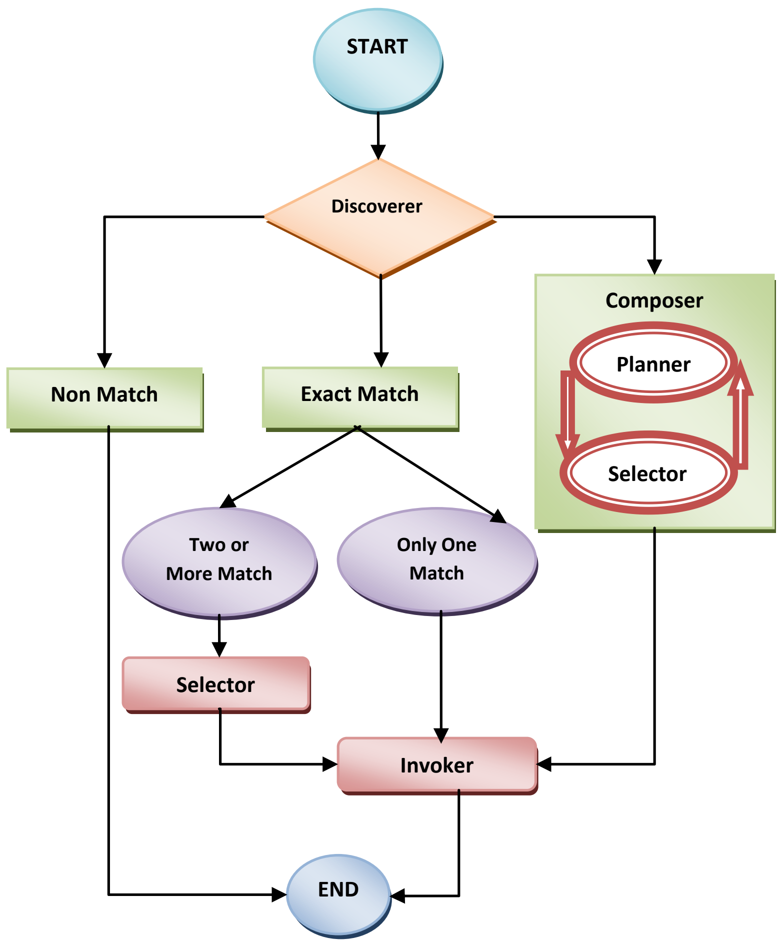

Figure 5.2 Flow chart of the proposed framework

5.3.3 Concrete View of the WSS Mechanism

In this section, the mechanism of the proposed framework is described in detail. Furthermore, the contributions of this research are presented in detail. This approach is divided into three stages: *pre-selection, selection* and *post-selection*.

5.3.3.1 Pre-selection

This stage concerns the process before the selection stage, namely, what should be prepared for the selection stage. For the selection service, the QoS weights and the rates of alternatives must be prepared. The QoS weights represent how each criterion is important in relation to the other criteria. The rates of alternatives represent the score of alternative services for each criterion. The methods employed by this research to provide the rates of alternatives are described below.

- ***Default Weight***

The proposed solution for the criteria weights is flexible. First, the default weights are defined by employing the AHP algorithm, as described in Section 2.3.1. Then, those QoS weights will be offered to the user. If the user is an expert user and has sufficient knowledge about the weighting of criteria, then she can modify the weights. Otherwise, the user takes the default weights, which are produced using AHP and the input data of expert researchers in the area of web services. This range of possibilities can improve the flexibility of the proposed approach. The researchers' opinions about each criterion are collected. Those data are input data for the AHP algorithm, and AHP provides the default weights. The details of this method and algorithm are described in Section 5.4.1.1.

❖ ***Rates of Alternatives***

After preparing weights of the criteria, it is necessary to provide the rates of alternatives. As discussed in Section 5.1, in the existing literature, some approaches use the data of SPs and other approaches use the data of expert DMs without consideration of the CLs of the SPs and the power of the DMs. In this approach, this issue is resolved by applying the *CL* and *power of the DM* factors on the SPs and DMs, respectively. To improve the accuracy of the data, the average of these two parameters is used. The algorithm of this method is described in Section 5.4.1.2. The *pre-selection* stage is shown below.

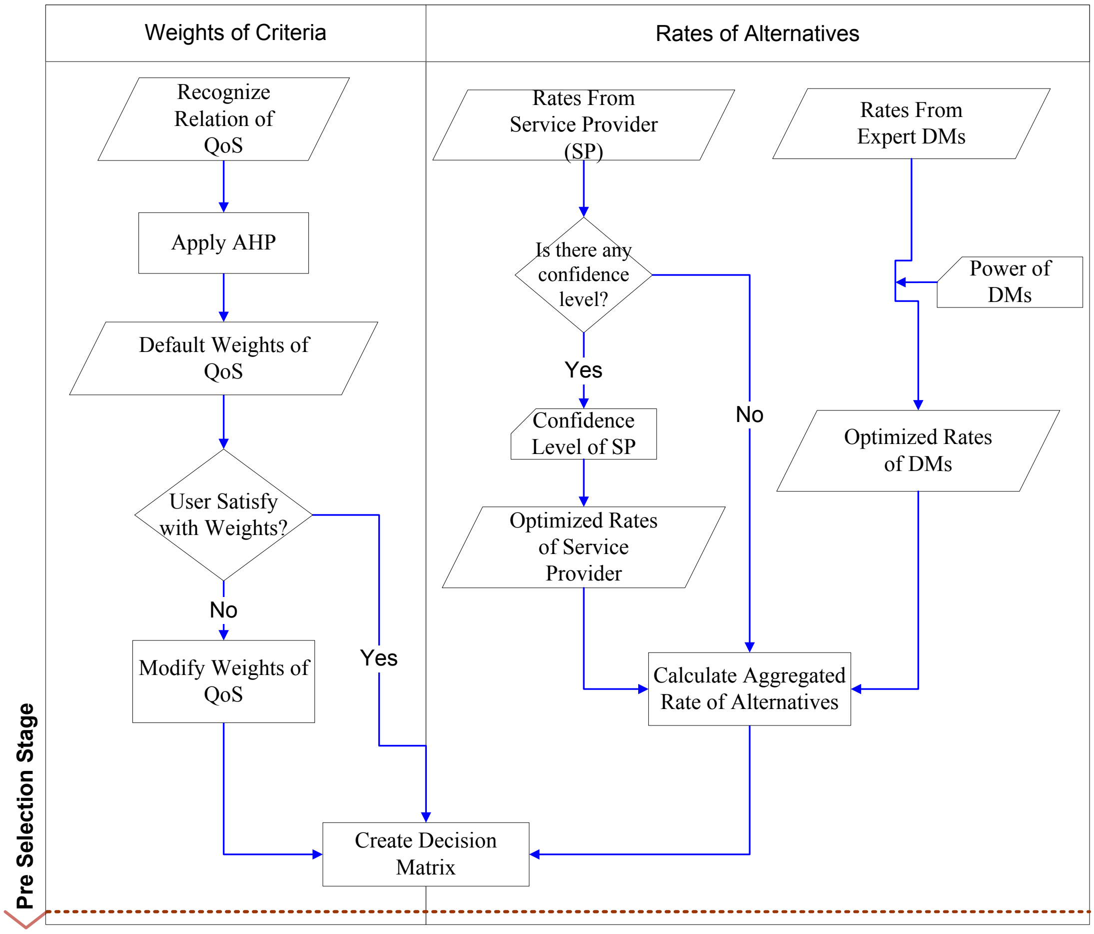

Figure 5.3 Flow chart of the pre-selection stage

5.3.3.2 Selection

The criteria weights and rate alternatives, which are prepared in the previous stage, will be the input data of the selection stage. In this stage, the decision matrix, which becomes accurate in the prior stages, is provided, and then, the enhanced VIKOR method is employed. The algorithm of this stage is described in Section 5.4.2. The *selection* stage is shown in Figure 5.3.

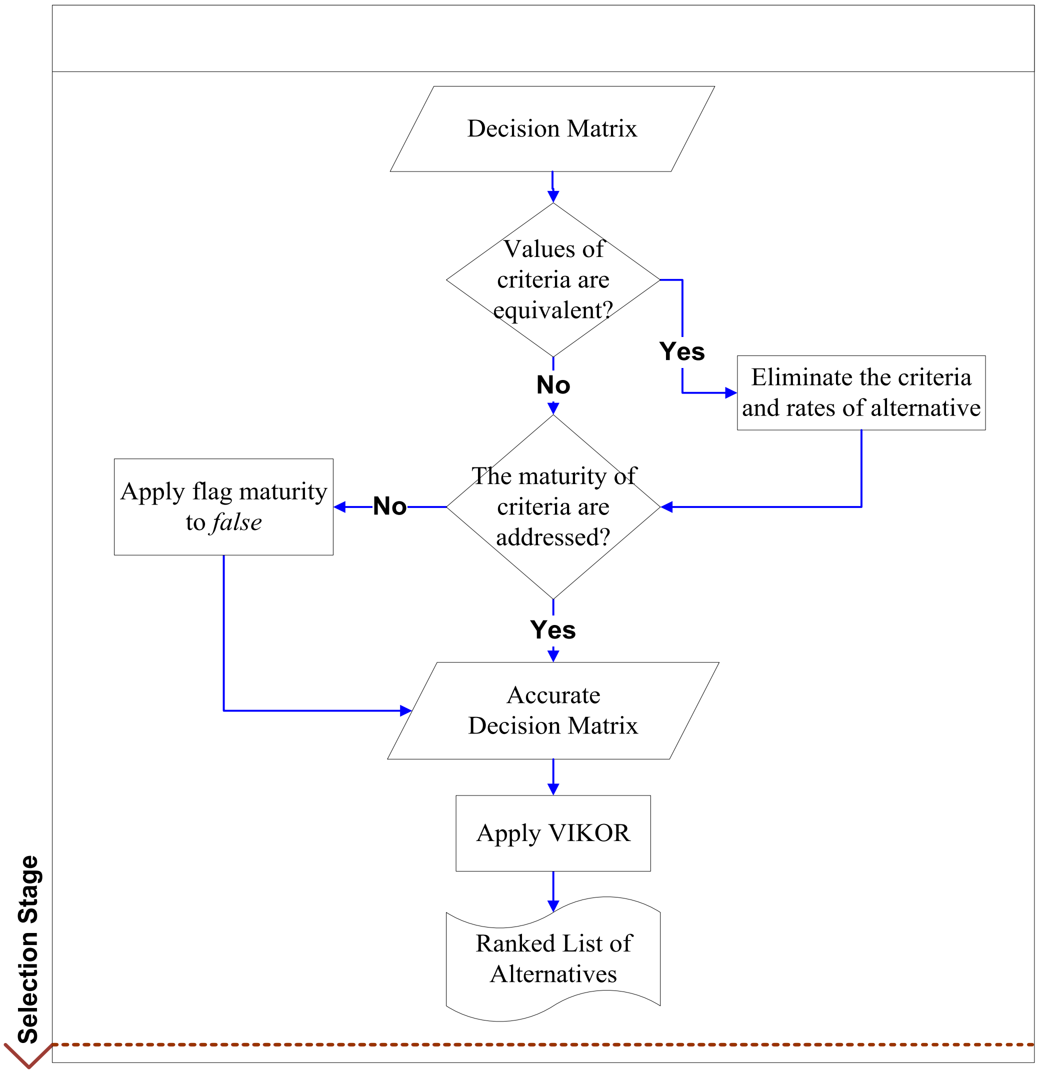

Figure 5.4 Flow chart of the selection stage based on the enhanced VIKOR algorithm

5.3.3.3 Post-selection

Once the selection is complete and the rank of alternative services is returned to the user, the approach updates the database of *default weights, the CL of the DMs* and *trust and reputation*. First, the data from the SPs is compared with the data from expert DMs. The CL of the SP is calculated by comparing those data. If the data of the SP are close to the data of the expert DMs, the CL of the SP will be increased; otherwise, it will be reduced. Furthermore, as in the proposed approach, there is a *reputation* criterion. The bank of trust and reputation is updated based on the feedback of users. The default weight bank is similarly updated based on the feedback from the service consumer. The flowchart of the post-selection stage is shown in Figure 5.5. The details and algorithm of this stage are described in Section 5.4.3.

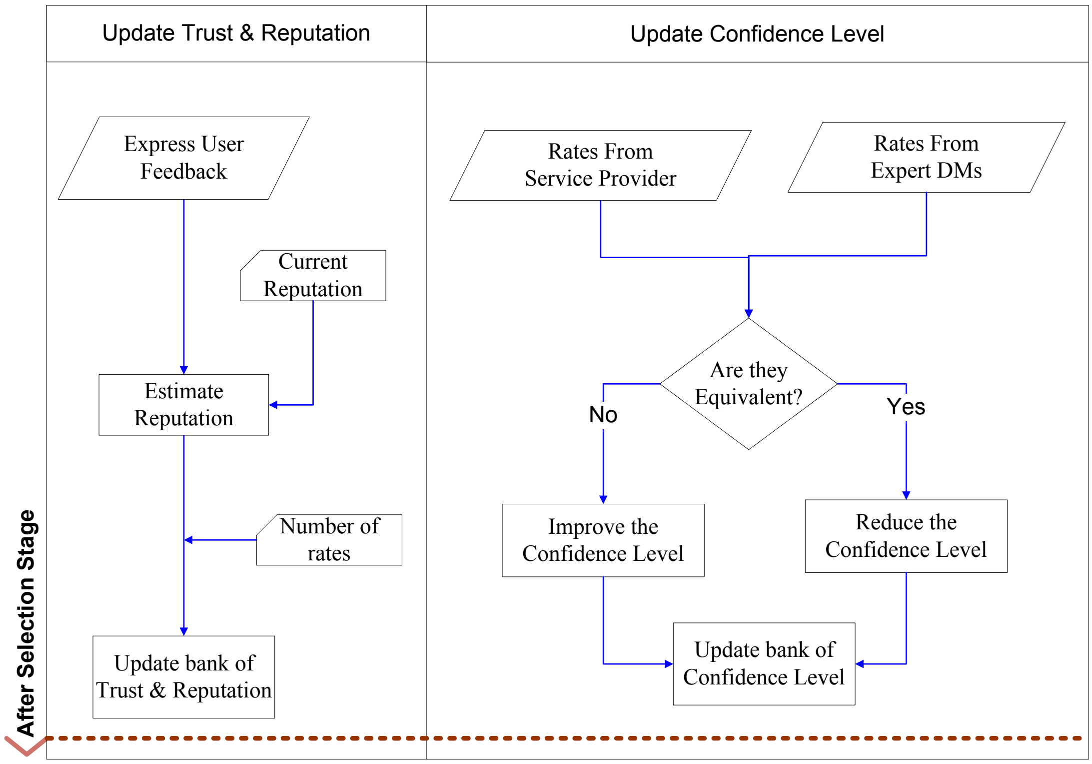

Figure 5.5 Flow chart of the post-selection stage

5.4 The Algorithmic Aspect of AMW

This section discusses the algorithms of AMW as the static aspect of the approach. These algorithms support the dynamic aspect and particularly the service selection mechanism described in Section 5.3.3. As described in that section, there are three stages in AMW for service selection. Each stage is discussed below.

5.4.1 Pre-selection

The pre-selection stage is divided to two parts: preparing default weights and collecting the data for the rates of alternatives. These parts are described below.

5.4.1.1 Default Weights

As described in Section 5.3.3.1, the solution of weighting criteria in AMW is flexible. Although AMW offers the default criteria weights, users can also modify them according to their preferences. In this section, the method for finding the default criteria weights is discussed. Finding the default weights of common QoS, such as *cost, performance* and *reliability,* is the aim of this section. In [19], some criteria are defined for the goal component of WSMO. Therefore, the QoS weights based on those criteria that are measurable and needed for service selection are defined. Those criteria, described in[19], are the following:

- ✓ *Performance (P)*: how fast a request of service can be completed.
- ✓ *Accuracy (A):* the accuracy of the result of service or the rate of error generated by the service.
- ✓ *Scalability (S):* the capability of the service to process more requests in a certain duration.

- ✓ *Financial (F):* also called *cost* or *price*; the cost and charge related to a service [45].

- ✓ *Reliability (R):* the ability of a service to achieve its requested tasks and functions.

- ✓ *Trust (T):* trustworthiness of the service, in contrast to *reputation,* which concerns the reputation of the SP.

- ✓ *Security (Se):* the ability of a service to provide authorisation, authentication, data encryption and traceability.

Apart from the above criteria, AMW also adds two more criteria, namely, *type of matching (TOM)* and *reputation of service (RS).* The *TOM* score ranges from zero to five, where zero is assigned to a no-match and five is assigned to an exact match. Service consumers cannot modify the weights of these two criteria because these criteria affect the factors of each service.

Consequently, there are seven QoS criteria and two offered criteria. The default criteria weights should be provided using the algorithm described in Figure 5.6. AMW uses the AHP method, described in Section 2.3.1, to determine the criteria weights. For this purpose, a matrix similar to the one shown below should be prepared to evaluate the criteria.

$$\begin{array}{c} \\ C_1 \\ C_2 \\ \vdots \\ C_n \end{array} \left[\begin{array}{cccc} C_1 & C_2 & \cdots & C_n \\ \hline X_{11} & X_{12} & \cdots & X_{1n} \\ X_{21} & X_{22} & \cdots & X_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ X_{i1} & X_{i2} & \cdots & X_{ij} \end{array}\right]$$

Algorithm 1: Define Weights of Criteria

```
Data: Matrix n*n of comparison of criteria MCom.
Result: default Weights of criteria WQoS.
begin
    β ←∅, is normalized matrix of MCom;
    Ω ←∅, where Ω is a set of tuples [Criteria,Weight];
    X_ij ∈ MCom;
    for i<=n do
        for j<=n do
            Sum_j = Sum_j + X_ij ;
        end
    end
    for i<=n do
        for j<=n do
            X'_ij = X_ij / Sum_j ;
            β = β ∪ X'_ij ;
        end
    end
    for j<=n do
        for X'_ij ∈ β do
            Weight_i = Weight_i + X'_ij ;
        end
        Ω = Ω ∪ [QoS_i, Weight_i];
    end
    WQoS ← Ω;
end
```

Figure 5.6 Defined criteria weights

The opinions of experts in the area of web services are gathered based on Table 2.1; the aggregated data are shown in Figure 5.7.

	P	A	S	F	R	T	Se	TOM	RS
P	1	2	5	3	5	6	4	1	1
A	1/2	1	4	2	4	5	3	1/2	1/2
S	1/5	1/4	1	1/3	1	1	1/2	1/5	1/5
F	1/3	1/2	3	1	4	5	4	1	1
R	1/5	1/4	1	1/4	1	1/2	1	1/3	1/3
T	1/6	1/5	1	1/5	2	1	1	1/3	1/3
Se	1/4	1/3	2	1/4	1	1	1	1/4	1/3
TOM	1	2	5	1	3	3	4	1	1
RS	1	2	5	1	3	3	3	1	1

Figure 5.7 Pair-wise comparing matrix

Based on the data expressed in the above matrix and Algorithm 1, the AHP method is as follows. First, the sum of each column is prepared using the following formula:

$$Sum_j = \sum_{i=1}^{n} (X_{ij})$$

$Sum_j =$

P	A	S	F	R	T	Se	TOM	RS
4.65	8.53	27	9.3	24	25.5	21.5	5.6	5.7

Then, the X_{ij} is divided bySum_j,

	P	A	S	F	R	T	Se	TOM	RS
P	0.22	0.23	0.19	0.33	0.21	0.24	0.19	0.18	0.18
A	0.11	0.12	0.15	0.22	0.17	0.20	0.14	0.09	0.09
S	0.04	0.03	0.04	0.04	0.04	0.04	0.02	0.04	0.04
F	0.07	0.06	0.11	0.11	0.17	0.20	0.19	0.18	0.18
R	0.04	0.03	0.04	0.03	0.04	0.02	0.05	0.06	0.06
T	0.04	0.02	0.04	0.02	0.08	0.04	0.05	0.06	0.06
Se	0.05	0.04	0.07	0.03	0.04	0.04	0.05	0.04	0.06
TOM	0.22	0.23	0.19	0.11	0.13	0.12	0.19	0.18	0.18
RS	0.22	0.23	0.19	0.11	0.13	0.12	0.14	0.18	0.18

After employing the $X'_{ij} = (X_{ij} \,/\, Sum_j)$ and $Weight_i = \sum_{j=1}^{n}(X'_{ij})$, the default criteria weights for assessing the web services are as follows:

$$Weight_i = \begin{matrix} P \\ A \\ S \\ F \\ R \\ T \\ Se \\ TOM \\ RS \end{matrix} \begin{bmatrix} 0.22 \\ 0.14 \\ 0.04 \\ 0.14 \\ 0.03 \\ 0.05 \\ 0.05 \\ 0.17 \\ 0.16 \end{bmatrix}$$

This matrix demonstrates that the performance criterion is more important than the other criteria. These weights are used in the next chapter once the service selection process is completed by AMW.

5.4.1.2 Rates of Alternatives

As described in Section 5.3.3.1, AMW provides the rates of alternatives in two ways: the SPs and expert DMs. The details and flowchart of how AMW provides the rate is also described above. In this section, the algorithm shown in Figure 5.8 is provided to support the pre-selection portion of the dynamic aspect.

The first time that the service of a certain SP is used, the CL of the SP is not recognised. Therefore, AMW trusts the SP and the rates of alternatives are captured via expert DMs with respect to the power of the DMs simultaneously. Finally, the aggregated rates of alternatives are provided by averaging those rates. After one selection time, the CL of the SP will be clarified, and from then on, the given rates of alternatives expressed by the SP are evaluated with respect to the CL of the SP. These details are shown in Algorithm 2.

Algorithm 2: Define Rates of Alternatives

```
Data: Rates of alternatives from SP: RSP.
Data: Rates of alternatives from expert DMs: RDM.
Result: Aggregated rates of alternatives RAgg.
begin
    α is confidence level of SP;
    β_{1,2,3} is power of expert DMs;
    X_ij = extractRates(RSP);
    Y_ij = extractRates(RDM);
    Ω ←∅, where Ω is optimized rates from SP;
    λ ←∅, where λ is optimized rates from DMs;
    if α <>∅ then
        for i<=n do
            for j<=m do
                X'_ij = α * X_ij ;
                Ω = Ω ∪ X'_ij ;
            end
        end
    end
    else
        Ω ← RSP;
    end
    for p<=3 then
        for i<=n do
            for j<=m do
                Y'_ij = β_p * Y_ij ;
                λ = λ ∪ Y'_ij ;
            end
        end
    end
    RAgg ←Average (Ω , λ);
end
```

Figure 5.8 Defined rates of alternatives

5.4.2 Selection

As discussed in Section 5.3.1.6, the central component of AMW is the selector and DM, where all prepared weights, rates of alternatives, the CL, the power of the DMs and the type of matching appear in this stage.

First, the default criteria weights, defined via Algorithm 1, pass to the service consumer. If a user wants to change the weights, she is allowed to do so. The final criteria weights and rates of alternatives, gathered via Algorithm 2, are sent to the enhanced VIKOR. The enhanced VIKOR is based on the VIKOR with improved features and eliminated shortcomings, as described in Section 5.3.1.6.

As shown in lines 8-20 of Algorithm 3, the first shortcoming of VIKOR is solved by eliminating criteria data when the values are equivalent in the decision matrix. As those criteria do not affect the service selection results, they are removed from the matrix.

Lines 21-30 attempt to solve the second shortcoming of VIKOR, described in Section 5.3.1.6. As an efficient result, the selected service must address the maturity of the criteria. Otherwise, the selected service is not the most appropriate, and the alternative service, which includes the second rank and addresses most of criteria, is selected.

Finally, the ranked list of services with respect to the user preferences is created. Then, the description of the service on the top of list is returned back as the result of AMW, and the service is passed for invocation. Because the AMW is dynamic, the description of the second best service in the ranked list is sent for invocation when the highest-ranking service is not available. The core of the selection algorithm is shown in Figure 5.9.

Algorithm 3: Web Service Selection

Data: Decision Matrix based on Algorithm1 & Algorithm2: DM_{n*m} *[Alternative, Weight]*.
Result: Best Service *BS*, Ranked Service list *RWS*.

```
begin
    X_ij = extractRates(DM_n*m);
    β ←∅, is normalized matrix of DM_n*m;
    flag ← true, is about checking the equality of rates;
    flagMaturity ← false, where flagMaturity is for controlling the maturity of zero rates;
    fmax_m=0, fmin_m=0, Sum_j=0;
    LQ_i ←∅, Sort list of Services;
    for j<=m do
        for i<=n do
            if X_ij <>X_i+1j then
                flag ← false;
            end
        end
        if flag = true then
            for k<=n do
                eliminate DM[Alternative_k];
            end
            eliminate DM[Weight_j];
        end
    end
    for i<=n do
        for j<=m do
            if X_ij = 0 then
                Counter = counter + 1;
            end
        end
        if  Counter > (m / 2) + 1 then
            flagMaturity_i= true;
        end
    end
```

```
for j<=m do
    fmax_j ← Max (X_ij )
    fmin_j ← Min (X_ij )
end
for i<=n do
    for j<=m do
        X'_ij =(fmax_j − X_ij ) / (fmax_j − fmin_j );
        β = β ∪ (DM[Weight_j] * X'_ij );
    end
end
for i<=n do
    for X'_ij ∈ β do
        Sum_j = Sum_j + X'_ij ;
    end
    S_i ← Sum_j ;
    R_i ← Max_j (X'_ij );
end
Smax ← Max_i (S_i);
Smin ← Min_i (S_i);
Rmax ← Max_i (R_i);
Rmin ← Min_i (R_i);
for i<=n do
    if Smax = Smin then
        Q_i = (R_i − Rmin ) / (Rmax − Rmin);
    end
    else if Rmax = Rmin then
        Q_i = (S_i − Smin ) / (Smax − Smin);
    end
    else
        Q_i = 0.5*((R_i − Rmin ) / (Rmax − Rmin)) + 0.5*((S_i − Smin ) / (Smax − Smin));
    end
end
LQ_i ← sort (Q_i, Ascending);
DQ = 1 / (n-1);
for i<=n do
    if (LQ_{i+1} − LQ_i > DQ) And (flagMaturity_i = false) then
        BS ← LQ_i;
        Exit Loop;
```

```
        end
    end
    RWS ← LQ_i;
    end
```

Figure 5.9 WSS based on enhanced VIKOR

5.4.3 Post-selection

Now, the CL of the SP is estimated, and the reputation service is estimated based on feedback given by the user about the service. The reputation of the service is separated from the reputation and the CL of the SP because a SP might have more than one service, and only a few of these services are good. It is not fair to recognise all services of a SP equally. Consequently, AMW employs two factors to distinguish services from SPs. These two algorithms are described in the following sections.

5.4.3.1 CL Estimator

The CL of the SP is evaluated by comparing the expressed rates of the SP with the given rates of expert DMs. Occasionally, the SP is trustable and the expressed rates are valid. In contrast, some SPs incorrectly advertise their services. In these circumstances, the rates of the SP and expert DMs are compared. If the rates are similar, the CL of the SP increases. Otherwise, AMW decreases the CL of the SP. Algorithm 4 estimates the CL of the SP.

Algorithm 4: Estimating Confidence Level of Service Provider

Data: Service Provider Alternative Rates SPrates,
Expert DM Alternative Rates DMrates.
Result: Confidence Level of Service Provider CLsp.

```
begin
    X_ij , extract data of SPrates_{n*m};
    X'_ij , extract data of DMrates_{n*m};
    Rsimilarity, Similarity ratio of data;
    Ω ←∅, Comparing score of rates;
    β, Summarize of Confidence Level;
    for i<=n do
        for j<=m do
            Rsimilarity = checkSimilarity(X_ij , X'_ij );
            if Rsimilarity>0 and Rsimilarity<30% then
                Ω = Ω ∪ - 1;
            end
            else if Rsimilarity>30% and Rsimilarity<70% then
                Ω = Ω ∪ 0.5;
            end
            else
                Ω = Ω ∪ 1;
            end
        end
    end
    for C_ij ∈ Ω do
        for i<=n do
            for j<=m do
                β = β +( β * (C_ij /10));
            end
        end
    end
    CLsp ← β;
end
```

Figure 5.10 The Algorithm of estimating CL of SP

5.4.3.2 Reputation Estimator

In this section, the estimation of the reputation of each service is described. AMW provides an algorithm to estimate the reputation of each service that is based on user feedback. After the selection process, the user feedback is gathered and the reputation is estimated based on the formula below.

$$Reputation = \sum_{i=1}^{n} \left(\frac{R_i}{n}\right)$$

Where n is the number of rates and R_i is the given rates for a certain service. If there are more users rating the service, the assurance of the reputation rate is higher, as shown in Algorithm 5. Once the reputation of a service is evaluated, the trust and reputation bank is updated with the new rate.

Algorithm 5: Estimating Reputation of Service

Data: Feedback of Service Consumer *USERf*,
List of Existing feedback of service *Lrep*.
Result: Updated Reputation of Service *REPnew*,
Updated List of feedback *LrepNew*.
begin
    $score_i$, extract data of *Lrep*;
    $\beta \leftarrow Lrep;$
    $\beta = \beta \cup USERf;$
    for i<=n do
        for $score_i \in Lrep$ do
            $Sum = Sum + score_i$;
        end
    end
    $REPnew \leftarrow Sum$;
    $LrepNew \leftarrow \beta$;
    end

Figure 5.11 The algorithm of estimating the reputation of a service.

5.5 Goal Generator

The aim of this section is the generation of the goal WSML file based on the expressed user preferences. For this purpose, it is first necessary to describe the format and structure of the goal in the WSML format.

5.5.1 Goal WSML File

A goal WSML file contains the capability, interfaces and non-functional properties. The capability describes the required functionality using logical expressions, the interfaces describe the interaction of services and non-functional properties express the annotation of the information. The aim of this section and the goal generator is to create the non-functional properties. Figure 5.12 describes a simple goal WSML file for capturing the requirements of a user [19].

```
Goal "http://example.org/bookGoal"
   nonFunctionalProperties
     dc#title hasValue "Example Book Buying Service"
     dc#description hasValue "Example Goal for adding items to a shopping cart"
     dc#price hasValue " Example price of service"
     dc#hasWeights hasValue req#DefinitionWeights
   endNonFunctionalProperties

   importsOntology "http://example.org/bookOntology"
    capability
     sharedVariables {?cartId , ?item}
     precondition
       definedBy
          ?cartId memberOf string and ?item memberOf book.
     postcondition
       definedBy
          forall ?cart ( ?cart [ id hasValue ?cartId ] memberOf cart implies
             ?cart [ items hasValue ?item]) .
     axiom DefinitionWeights
          definedBy
             hasWeight(dc#hasPrice, 60).
```

Figure 5.12 A sample of goal in WSML format.

As shown in the above sample, the section for non-functional properties begins with "*nonFunctionalProperties*" and is closed by "*endNonFunctionalProperties*". Each non-functional property includes the name and value of certain non-functional properties with the following structure:

"Name of nfp"***hasValue***"Value of nfp"

The name, value and weight of the non-functional properties are required. As shown in Figure 5.12, the criteria weights can be declared using an axiom. The phrase weights are declared in part of the non-functional properties. Then, in the axiom, the weights appear as in the following example:

hasWeight(dc#hasPrice, 60)

This example of an axiom describes that the weight for price, as a non-functional property, is 60. The remaining non-functional properties can be implemented with a structure similar to that employed for price. Based on the definitions and descriptions in the previous section, the manner in which AMW generates the goal WSML file based on preferences and default weights of the criteria is described.

5.5.2 Flowchart of the Goal Generator

This section illustrates the process of the goal generator using a flowchart. First, the default criteria weights, defined in Section 5.4.1.1, are expressed to the user, and the user can modify them. Those criteria and two additional criteria, *TOM* and *RS*, are finalised. Then, the generator checks whether each criterion is greater than zero. Weights changed to zero by the user should not be used because a value of zero means that the user does not care about that criterion. In the next step, the final list of criteria with related weights are passed on for conversion to the WSML file. For each criterion, one phrase is generated in the section of *non-functional properties*. Similarly, for any criteria weights greater than zero, one phrase is generated as an axiom.

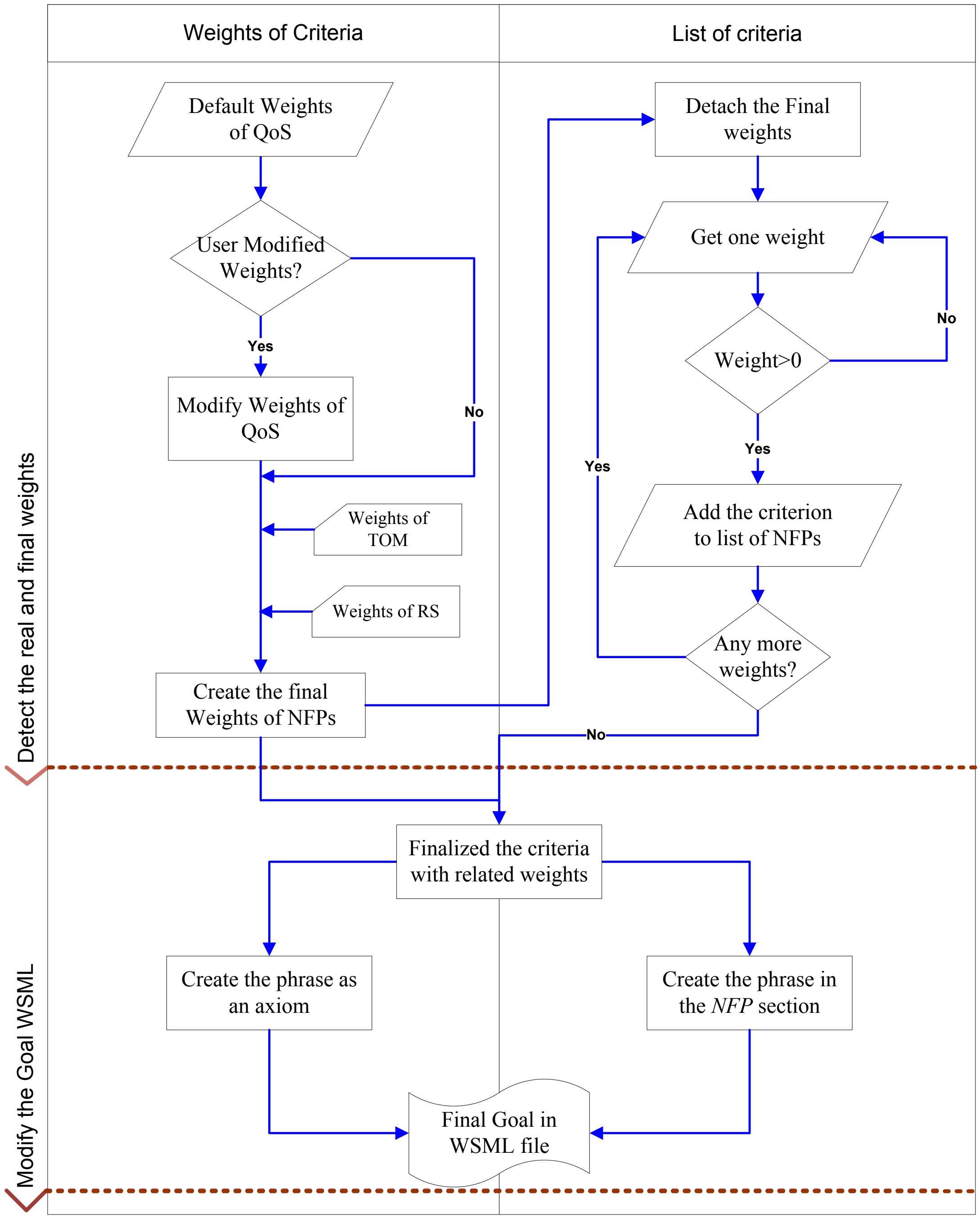

Figure 5.13 Flowchart of the goal generator.

5.5.3 Algorithm of the Goal Generator

In this section, the algorithm of the goal generator is described. This algorithm is based on the flowchart described in the previous section. The goal generator algorithm that is part of AMW is shown below.

```
Algorithm 6: Goal Generator

Data:   Goal WSML file GoalF,
List of user preferences LPre.
Result: Goal WSML file with NFP WSMLG.
begin
    Ω ←∅, where Ω is a pre WSML for new goal;
    β = extract (LPre), where β is a set of triple [criteria, value, weight];
    λ , is variable of define weights;
    divide the WSML to GoalF[header, body] ;
    Ω = Ω ∪ GoalF[header];
    addTagΩ "nonFunctionalProperties";
    for criteria ∈ β do
        addTagΩ β [criteria] ∪ "hasValue" ∪ β [value];
    end
    addTagΩ "endNonFunctionalProperties";
    addTagΩ "hasWeights hasValue λ"
    Ω = Ω ∪ GoalF[body];
    addTagΩ "axiom λ"
    addTagΩ "definedBy"
    for criteria ∈ β do
        addTagΩ "hasWeights" ∪ β [criteria] ∪ β [weight];
    end
    WSMLG ← Ω
    end
```

Figure 5.14 Goal generator algorithm.

5.6 AMW Prototype Implementation

In this section, the implementation of the AMW prototype is described based on the architecture and algorithmic aspects described in Sections5.3 and 5.4. The aim of the implementation is to facilitate the evaluation of this research experimentally. As the basis of this research is WSMO and its execution environment (WSMX) is implemented in Java, the prototype is also implemented in Java. Therefore, in the future, this prototype can be integrated with WSMX with only minor modifications.

This component, similar to the architecture and algorithmic components, is divided into three subsections: *pre-selection*, *selection* and *post-selection*. Several important interfaces of the prototype are shown in this section. The implementation of the prototype based on the above division is described below.

5.6.1 Pre-selection

This subsection involves preparing the default criteria weights and the input data for service selection. The goal generator is also considered, as the first step of WSS is to have a goal file. In the implementation, the preparation of default weights and the goal are presented in as a single "*Generate Goal*" module because having the default criteria weights is a precondition of generating the goal. The preparation of rates of alternatives, which include such concepts as CL and the power of the DMs, are implemented in the "*Add Service*" module. These two modules are discussed in detail below.

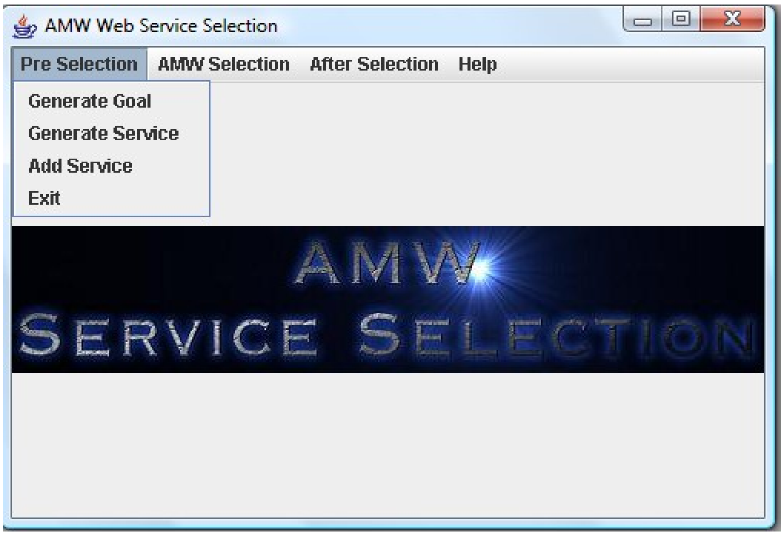

Figure 5.15 The main GUI of the AMW prototype.

5.6.1.1 Goal Generation

This module is implemented based on Algorithms 1 and 6. First, Algorithm 1, which finds the default criteria weights, is implemented. As discussed in Section 5.4.1.1, the basis of finding default weights is the AHP method described in Section 2.3.1. Thus, this algorithm is used to determine the default weights and is implemented via the "*defaultWeights*" class. Goal generation is the other process of this module, which translates the user preferences to the WSML format. The procedure and steps of this process are described in Algorithm 6. To implement this part, the "*GoalGenerator*" class is provided to collect the preferences of the user and translate them into the WSML format. These preferences are translated in the non-functional properties part (*NFP*), and an *Axiom* is inserted at the end of WSML to define the criteria weights. Now, the criteria weights may be default or user-modified weights. Finally, this WSML format is added in the specified goal file. A screenshot of this module is shown below.

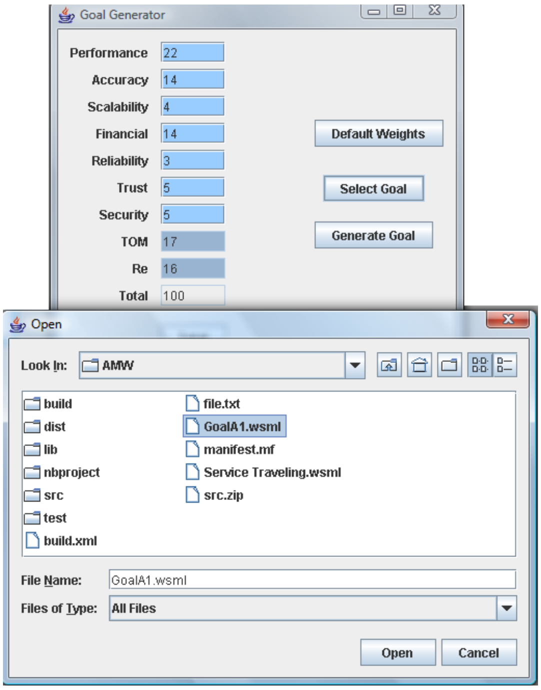

Figure 5.16 The goal generator used to translate user preferences to WSML format.

5.6.1.2 Add Service

This module is implemented based on Algorithms 2 and 4. Although Algorithm 4 is presented in the *post-selection* part, it is required for calculating the rates of alternatives upon adding the service to the repository. Therefore, Algorithm 4 is also presented in this section.

To add a service, the service is first chosen. Then, all required and suitable data are extracted. These data include the service name, SP, current CL of the SP and values of non-functional properties. These data extraction is implemented in the "*grabValue*" sub-class, which reads the WSML service file and extracts the CL of the SP from the repository. The remaining data are extracted from the service WSML file.

To achieve more accurate input data (rates of alternatives), two factors are considered, the CL of the SP and the power of the DMs, which are applied in Algorithm 2. If the service could not be found in the repository, it should be added. Otherwise, the service already exists in the repository. For the former case, the expressed rates by the expert DMs with respect to the power of the DMs are optimised, and then, these rates are integrated with the expressed rates by the SP with respect to the CL of the SP. The "*addServiceToRepository*" sub-class adds the service with aggregated or accurate data to the repository.

Finally, after adding the service to the repository, the new CL of the SP should be estimated, which is performed by comparing the expressed rates by the SP and expert DMs. The formula is developed in Algorithm 4. This algorithm is also implemented in the "*calculateCL*" sub-class. Comparing the rate determines whether the current CL should be increased, decreased or left unchanged. The interface of "*Add Service*" is shown in Figure 5.17.

5.6.2 Selection

This section is the central element of AMW, and the related formula is presented in Algorithm 3.Therefore, the implementation in this section is based on that algorithm, which is described in Section 5.4.2. Based on Algorithm 3, the input of the algorithm is a decision matrix, which involves the criteria weights and rates of alternatives. As mentioned earlier, the default criteria weights are calculated using Algorithm 1 and implemented by the "*defaultWeights*" class. The accurate rates of alternatives are aggregated by Algorithm 2 and implemented by the "*addServiceToRepository*" class. The outputs of Algorithms 1 and 2 are the input of Algorithm 3. This algorithm is implemented using the "*serviceSelection*" class.

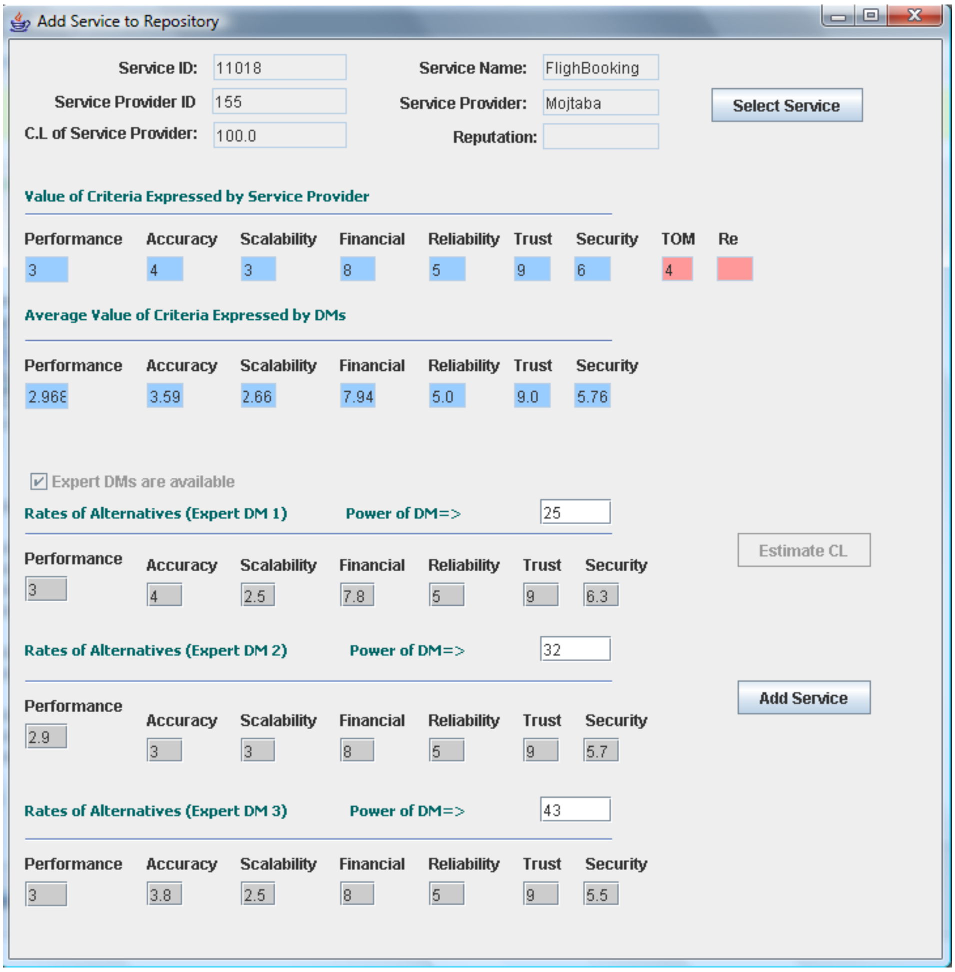

Figure 5.17 Extract and integrate accurate data and add it to the repository.

The default weights are recommended to the user. If the user is an expert and has knowledge of the criteria weighting, the user will modify the default weights. Then, the user selects the required WSML goal file, which is generated automatically, as described in the previous section. Next, the rates of discovered services are gathered from the repository. The decision matrix is provided based on the rates of alternatives and the criteria weights.

The decision matrix is passed to Algorithm 3 for ranking and to select the appropriate service. As explained in Algorithm 3, after equal criteria values are eliminated, the matrix is sent to the VIKOR method. Then, the rates are normalised. The services are ranked based on the normalised matrix. Next, the ranked list of services is offered to the service consumer, and the service consumer selects one. Users will likely select the service at the top of the list, but they can also select the other services based on their preferences. Finally, two additional highly ranked services are considered as alternatives for the selected service. Below is a screenshot of the AMW service selection.

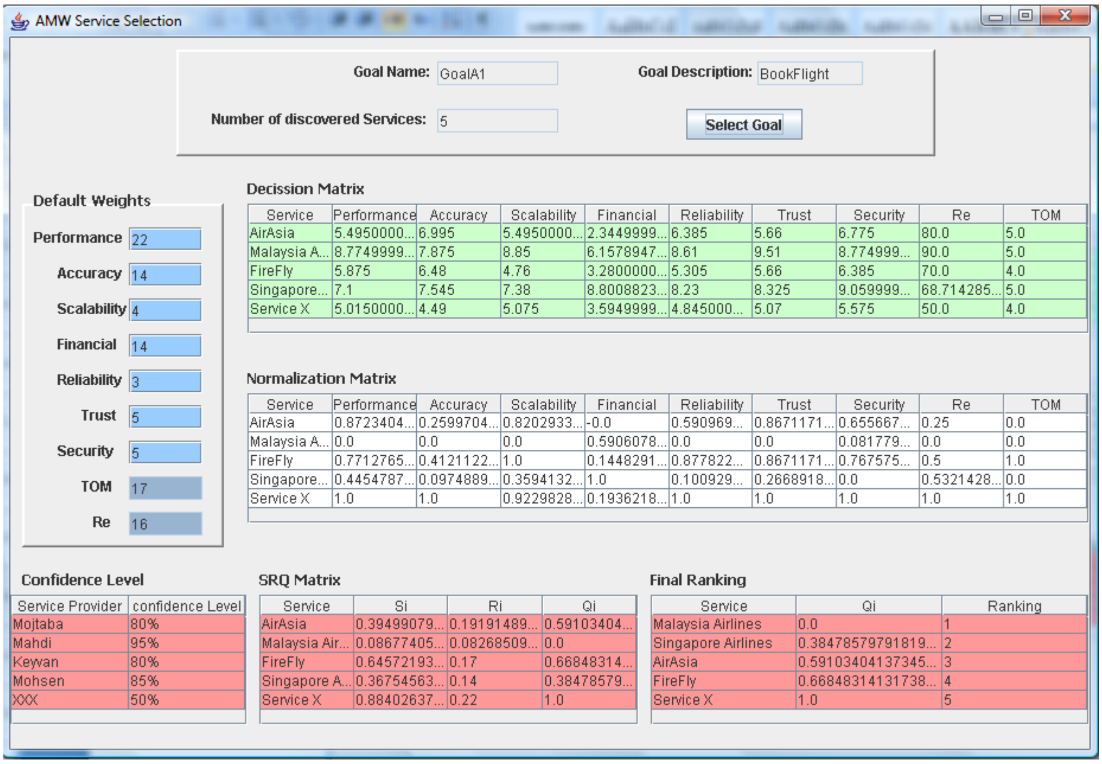

Figure 5.18 Screenshot of the AMW service selection.

5.6.3 Post-selection

Post-selection is the final step, and it is performed after the users select the service. The basis of this section is Algorithm 5, which is described in Section 5.4.3.2. The implemented class for this algorithm is "*Feedback*". In this section, the reputation of the service is estimated. For this purpose, users express their feedback for a certain service, giving it a rank between zero and 100. If the new service rank is averaged with the prior reputation, the result would not be accurate. Suppose that after 100 times, the reputation of a service is 95, and the service takes 25 as the new rank. Then, the new reputation of service would become 60. To avoid this inaccurate reputation, all user feedbacks is gathered, and the average reputation is estimated. For the given example, the new reputation becomes 94.3. Therefore, the feedback of one user could not significantly affect the reputation of a service if the reputation was achieved after several uses of the service.

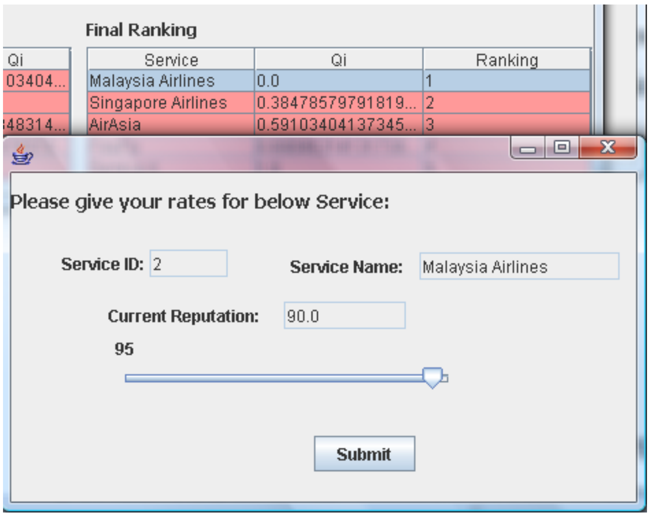

Figure 5.19 Estimated reputation of the service after AMW service selection.

5.7 Summary

This chapter describes the AWM approach to supporting WSS. First, the limitation of the current approach is specified in Section 5.1. Second, the AMW approach is declared, and the *architectural* and *algorithmic* aspects are described. The architectural aspect, described in Section 5.3, attempts to explain the proposed approach using a flowchart, diagram and framework, whereas the algorithmic aspect, described in Section 5.4, attempts to present the algorithm of each component that supports AMW and is described in the architectural aspect. Finally, the goal generator, which is part of AMW, is introduced.

The proposed solution for providing flexible and automated service selection involves the application of a flexible framework and proficient MCDM method. AMW employs the VIKOR method with enhanced features, which can resolve the shortcomings of the current approaches considered in Section 5.3.1.6. In addition to improving the automation of service selection, AMW uses a generator to convert user preferences into the format of the goal WMSL file. In the next chapter, using a case study, the manner in which AMW can select the most appropriate service with respect to user preferences and expert DMs is demonstrated.

Finally, to facilitate the evaluation of this research based on the experiments, one prototype is implemented based on the architecture and algorithm of the above approach. Similar to the architecture and algorithmic aspects, the prototype is also separated into three parts. Each part is described in detail, and the advantages of the prototype are presented. In the next chapter, the evaluation of the AMW is described and the applicability of the proposed approach for WSS is proven.

CHAPTER 6

EVALUATION OF THE PROPOSED APPROACH

This chapter describes the evaluation of the AMW approach. The aim of this section is to identify the types of problems that AMW can solve that cannot be solved by the existing approaches. The proposed approach must be validated analytically. Specifically, the conditions in which each of the existing approaches fails must be explained and specified, and the reasons for which AMW worked correctly also must be explained. The explanations are validated by demonstrations of each approach applied to specific examples that illustrate each of the conditions identified in the analysis. The results demonstrate how the features of AMW can affect the accuracy of WSS.

6.1 Case Study

The aim of this section is to present the features of AMW. For this purpose, two simulated case studies are used as applications of AMW. Because one case study is not able to cover all of the features of AMW, this research employs two different case studies: *flight booking (case study 1)* and *hotel reservation (case study 2)*. In both case studies, different scenarios are employed to demonstrate the influence of AMW on WSS. The possible states of the AMW approach are shown in Table 6.1.

Table 6.1 Possible states of the AMW approach

State	Description	State	Description
A1	Default Weights	***A2***	Modified Weights
B1	Rates from a SP with a CL	***B2***	Rates from a SP without a CL
C1	Aggregated rates from the SP and DMs	***C2***	Normal rates from the SP
D1	Values are different for all alternatives	***D2***	Values of a criterion are all identical
E1	The maturity of the criteria are addressed	***E2***	The maturity of the criteria are not addressed
F1	Rates of the SP are the same as those of the expert DMs	***F2***	Rates of the SP are different than the rates of the expert DMs

Based on the possible states described in the above table, 30 scenarios can be defined; however, most of the scenario results are the same, and thus, there is no benefit to describe them. This section lists all of the features of AMW for which the below questions must be answered. The questions are as follows: What is the result of AMW with...

1) default weights and modified weights?
2) respect to a CL and without a CL?
3) aggregated rates and normal rates?
4) different rates and similar rates?
5) the maturity of criteria addressed and not addressed?
6) similar and different rates of the SP and expert DMs?

The above questions can be answered by the six scenarios described in Table 6.2. The achievable scenarios are shown in the table below. Furthermore, the case study to which AMW is applied is specified.

Table 6.2 Achievable scenarios and assigned case study

Scenario	Situation	Description	Case Study
1	*A1+B1+C1+D1+E1+ F1&F2*	Basic scenario for AMW with all features	Case Study 1
2	*A2+B1+C1+D1+E1+ F1&F2*	Modified weights	
3	*A1+B2+C1+D1+E1+ F1&F2*	Aggregated rates without CL	
4	*A1+B1+C1+D1+E1+F1&F2*	Basic scenario for AMW with all features	Case Study 2
5	*A1+B1+C2+D1+E1*	Normal rates from the SP	
6	*A1+B1+C1+D2+E2*	Criteria weights have the same value and the maturity of criteria is neglected	

The first question will be answered by comparing the results of scenarios 1 and 2, and the second question will be answered by comparing the results of scenarios 1 and 3. The third question will be answered by comparing the results of scenarios 4 and 5, whereas questions 4, 5 and 6 are answered by comparing the results of scenarios 4 and 6. In Section 6.5, the results of all scenarios are compared with respect to the above description. Below, AMW is used in both case studies with the scenarios defined in the above table.

6.1.1 Flight Booking

In this section, an example is used to illustrate the proposed approach and demonstrate how AMW is implemented in service selection. First, the case study is described. Then, the scenarios defined in Table 6.2 are used. The first case study concerns flight booking web services. To search for flight booking services, the user must express functional and non-functional properties. Functional properties are used in the discovery stage, whereas non-functional properties are employed in the service selection stage.

Suppose that a user wants to find a service for booking a flight. In this case, given such requirements as the type of service, origin, destination, number of seats and date are considered as functional properties, whereas price, availability of service and performance are considered as non-functional properties. The system searches the appropriate services for flight booking services, and a number of services are discovered. Then, AMW selects the appropriate service based on non-functional properties. In the remainder of this section, the scenarios defined in Table 6.2 are used to demonstrate the applicability of AMW for service selection.

6.1.1.1 Scenario 1

In this scenario, a user is looking for a flight booking service from KL to Singapore on 12 December 2012. This scenario attempts to show all features of AMW in the standard situation. The default weights are used as criteria weights; the rates of alternatives are collected from the SP with respect to the SP's CL. Furthermore, those rates are aggregated with the rates from the expert DMs. There are no similar values for a certain criterion, and the maturity of the criteria is considered by the service alternative. Moreover, the rates from the SP are similar to the rates from the expert DMs.

After discovering all airline services, five airline services are available for the requested query. The relationship between the criteria and alternatives based on the description of each criterion is shown below.

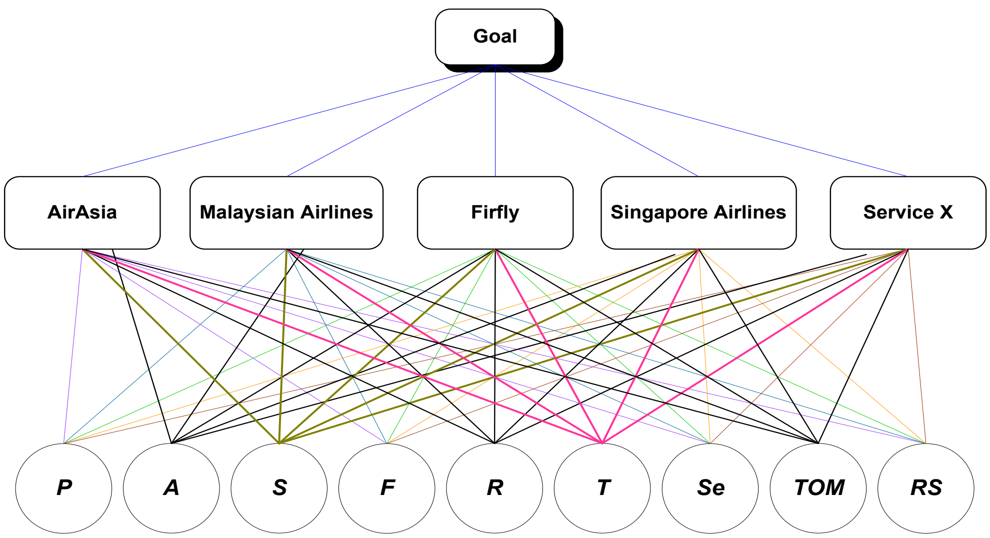

Figure 6.1 Relationship between the criteria and services.

As described in Chapter 5, AMW includes three stages: *pre-selection*, *selection* and *post-selection*. These stages are described below.

❖ ***Pre-selection***

This section follows the flowchart described in Figure 5.3. In this stage, the decision matrix is created. This matrix includes *n* rows and *nine* columns, where *n* refers to the number of alternatives and *nine* refers to the criteria.

In the first step of pre-selection, the criteria weights and rates of the alternatives are provided. As in this scenario, the default weights are used, and the data provided by AMW in Section 5.4.1.1are used as the default criteria weights:

$$Weight_i = \begin{bmatrix} P & A & S & F & R & T & Se & TOM & RS \\ 0.22 & 0.14 & 0.04 & 0.14 & 0.03 & 0.05 & 0.05 & 0.17 & 0.16 \end{bmatrix}$$

In the second step, the rates of these services are collected from the SP with respect to the seven criteria defined in Section 5.4.1.1. As the rates of *TOM* and *RS* are not collected from the SP, those data are gathered from the discovery part and the trust and reputation database, respectively. The rates (0-10) from the SP are as shown in Table 6.3.

Table 6.3 Rates of the alternatives from the SP

QoS / *Alternative*	*P*	*A*	*S*	*F*	*R*	*T*	*Se*
AirAsia	6	8	6	2	7	6	8
Malaysia Airlines	9	8	9	6	9	10	9
FireFly	7	7	5	3	6	6	7
Singapore Airlines	8	8	8	8	9	9	10
Service X	9	8	9	2	9	9	10

This rate should be considered by the CL for each service. As there is a database in AMW for the CL of the SP, those data are extracted. The CLs for the above services are shown in Table 6.4.

Table 6.4 Confidence level of service providers

Airline Service	***CL***
AirAsia	*80%*
Malaysia Airlines	*95%*
FireFly	*75%*
Singapore Airlines	*85%*
Service X	*10%*

Table 6.5 Rates of the alternatives from the SP with a CL

QoS / ***Alternative***	**P**	**A**	**S**	**F**	**R**	**T**	**Se**
AirAsia	*4.8*	*6.4*	*4.8*	*2.5*	*5.6*	*4.8*	*6.4*
Malaysia Airlines	*8.55*	*7.6*	*8.55*	*6.32*	*8.55*	*9.5*	*8.55*
FireFly	*5.6*	*5.6*	*4*	*3.75*	*4.8*	*4.8*	*5.6*
Singapore Airlines	*6.8*	*6.8*	*6.8*	*9.41*	*7.65*	*7.65*	*8.5*
Service X	*4.5*	*4*	*4.5*	*4*	*4.5*	*4.5*	*5*

Above the rates are shown with respect to CL. In the third step of pre-selection, the rates are also collected from expert DMs. The rates of three expert DMs are shown in Table 6.6.

Table 6.6 Rates of alternatives from the expert DMs

QoS / ***Alternative***		***P***	***A***	***S***	***F***	***R***	***T***	***Se***
d_1	***AirAsia***	*7.8*	*6.5*	*2.5*	*7*	*6.5*	*7.5*	*7.8*
	Malaysia Airlines	*8*	*9*	*6*	*8.5*	*9.5*	*9*	*8*
	FireFly	*7.5*	*5.5*	*2.5*	*5.5*	*6.5*	*7*	*7.5*
	Singapore Airlines	*8*	*7.5*	*8.5*	*8.5*	*9*	*9*	*8*
	Service X	*5*	*5.5*	*3.5*	*5.5*	*5*	*6*	*5*
d_2	***AirAsia***	*8*	*6*	*2*	*7*	*6*	*8*	*8*
	Malaysia Airlines	*8.5*	*9.5*	*6*	*8.5*	*9*	*9*	*8.5*
	FireFly	*7*	*5*	*3*	*6*	*6*	*7*	*7*
	Singapore Airlines	*9*	*8.5*	*8*	*9*	*9*	*10*	*9*
	Service X	*5.5*	*6*	*3*	*5*	*5.5*	*6.5*	*5.5*
d_3	***AirAsia***	*7*	*6*	*2*	*7.5*	*7*	*6*	*7*
	Malaysia Airlines	*8*	*9*	*6*	*9*	*10*	*9*	*8*
	FireFly	*7.5*	*6*	*3*	*6*	*7*	*7.5*	*7.5*
	Singapore Airlines	*8*	*8*	*8*	*9*	*9*	*10*	*8*
	Service X	*4.5*	*5.5*	*3*	*5*	*6.5*	*6*	*4.5*

However, these rates must be considered with the power of the DMs.

Table 6.7 Power of DMs

Decision Maker	*CL*
d1	*38%*
d2	*29%*
d3	*33%*

The rates from the DMs with the power of the DMs are then aggregated, as shown in Table 6.8.

Table 6.8 Rates of the alternatives from the expert DMs with power of the DMs

QoS / *Alternative*	P	A	S	F	R	T	Se
AirAsia	*6.19*	*7.59*	*6.19*	*2.19*	*7.17*	*6.52*	*7.15*
Malaysia Airlines	*9*	*8.15*	*9.15*	*6*	*8.67*	*9.52*	*9*
FireFly	*6.15*	*7.36*	*5.52*	*2.81*	*5.81*	*6.52*	*7.17*
Singapore Airlines	*7.4*	*8.29*	*7.96*	*8.19*	*8.81*	*9*	*9.62*
Service X	*5.53*	*4.98*	*5.65*	*3.19*	*5.19*	*5.64*	*6.15*

In the last step of pre-selection, the average of the rates from the SP and expert DMs are provided. Below, the final rates with respect to Algorithm 2 of AMW are shown.

Table 6.9 Aggregated rates of the alternatives

QoS / *Alternative*	P	A	S	F	R	T	Se
AirAsia	*5.5*	*7*	*5.5*	*2.35*	*6.39*	*5.66*	*6.78*
Malaysia Airlines	*8.78*	*7.88*	*8.85*	*6.16*	*8.61*	*9.51*	*8.78*
FireFly	*5.88*	*6.48*	*4.76*	*3.28*	*5.31*	*5.66*	*6.39*
Singapore Airlines	*7.1*	*7.55*	*7.38*	*8.8*	*8.23*	*8.33*	*9.06*
Service X	*5.02*	*4.49*	*5.08*	*3.6*	*4.85*	*5.07*	*5.58*

Two additional criteria and their rates are added. These criteria are *TOM* and *RS*, which are extracted from the AMW databases. The score of *TOM* ranges from zero to five, whereas the score of *RS* ranges from zero to 100.

Table 6.10 Final rates of the alternatives with *TOM* and *RS*

QoS / *Alternative*	P	A	S	F	R	T	Se	TOM	RS
AirAsia	5.5	7	5.5	2.35	6.39	5.66	6.78	5	80
Malaysia Airlines	8.78	7.88	8.85	6.16	8.61	9.51	8.78	5	90
FireFly	5.88	6.48	4.76	3.28	5.31	5.66	6.39	4	70
Singapore Airlines	7.1	7.55	7.38	8.8	8.23	8.33	9.06	5	80
Service X	5.02	4.49	5.08	3.6	4.85	5.07	5.58	3	50

❖ *Selection*

This section follows the flowchart described in Figure 5.4. In this section, the decision matrix provided in the previous section is given to Algorithm 3 of AMW to select the appropriate service based on the data provided in the prior section.

Step 1. The best (f_j^*) and worst (f_j^-) values for each column, shown in **Table 6.11**, are located.

Table 6.11 Best and worst values for all criteria

Criteria	P	A	S	F	R	T	Se	TOM	RS
f_j^*	8.78	7.88	8.85	2.35	8.61	9.51	9.06	5	90
f_j^-	5.02	4.49	4.76	8.8	4.85	5.07	5.58	4	50

As financial is a negative criteria, f_j^* is the minimum of the values, whereas f_j^- is the maximum of the values.

Step 2. Because there are positive and negative criteria and the criteria are not within the same scale, the matrix must be normalised by applying linear normalisation. The normalised matrix is shown in Table 6.12.

Table 6.12 Normalised decision matrix

QoS / *Alternative*	P	A	S	F	R	T	Se	TOM	RS
AirAsia	*0.87*	*0.26*	*0.82*	*0*	*0.59*	*0.87*	*0.66*	*0*	*0.25*
Malaysia Airlines	*0*	*0*	*0*	*0.59*	*0*	*0*	*0.08*	*0*	*0*
FireFly	*0.77*	*0.41*	*1*	*0.14*	*0.88*	*0.87*	*0.77*	*1*	*0.5*
Singapore Airlines	*0.45*	*0.1*	*0.36*	*1*	*0.1*	*0.27*	*0*	*0*	*0.25*
Service X	*1*	*1*	*0.92*	*0.19*	*1*	*1*	*1*	*1*	*1*

Step 3. The appropriate matrix used to compare S_i and R_i can be determined based on the normalised matrix.

Table 6.13 Values of S_i and R_i

Alternative	S_i	R_i
AirAsia	*0.39*	*0.19*
Malaysia Airlines	*0.08*	*0.08*
FireFly	*0.65*	*0.17*
Singapore Airlines	*0.31*	*0.14*
Service X	*0.89*	*0.22*

Step 4. In this step, S^-, S^+, R^- and $R^-$are calculated to specify the index values Q_i. The maximum and minimum values ofS_i and $R_i$are as follows:

$$S^- = 0.08, S^+ = .89 \text{ and } R^- = 0.08, R^+ = 0.22$$

At this time, Q_i, which is the index value used to rank the alternatives, can be determined.

$$Q_i = \begin{matrix} AirAsia \\ Malaysia\ Airlines \\ FireFly \\ Singapore\ Airlines \\ Service\ X \end{matrix} \begin{bmatrix} 0.584 \\ 0 \\ 0.673 \\ 0.356 \\ 1 \end{bmatrix}$$

Step 5. The three lists, S_i, R_i and Q_i, are ranked in ascending order. The minimum value has the highest score:

Table 6.14 Ranking of the alternatives

Alternative	*Rank of*		
	S_i	R_i	Q_i
AirAsia	*3*	*4*	*3*
Malaysia Airlines	*1*	*1*	*1*
FireFly	*4*	*3*	*4*
Singapore Airlines	*2*	*2*	*2*
Service X	*5*	*5*	*5*

Step 6. In this step, the services are ranked based on Q_i,S_i and R_i. The service for *Malaysia Airlines* is the best option with respect to the QoS criteria, whereas *Service X* is the worst option. The final ranking list based on scenario 1 is as follows:

$$\boldsymbol{MalaysiaAirlines > Singapore\ Airlines > AirAsia > FireFly > Service\ X}$$

- ***Post-selection***

This section follows the flowchart described in Figure 5.5. First, the CL of the SP is estimated. Then, the reputation of the service is provided based on user feedback.

The CL is estimated based on Algorithm 4 of AMW. First, the similarity of the data are calculated by comparing the data of the SP and expert DMs.

Table 6.15 Calculated similarity rates

QoS / ***Alternative***	**P**	**A**	**S**	**F**	**R**	**T**	**Se**
AirAsia	*1*	*1*	*1*	*1*	*1*	*1*	*1*
Malaysia Airlines	*1*	*1*	*1*	*1*	*1*	*1*	*1*
FireFly	*1*	*1*	*1*	*1*	*1*	*1*	*1*
Singapore Airlines	*1*	*1*	*1*	*1*	*1*	*1*	*1*
Service X	*0.5*	*0.5*	*0.5*	*0.5*	*0.5*	*0.5*	*0.5*

Then, the average similarity and updated CL of the SP are estimated.

Table 6.16 Estimated CL

Alternative	*Average*	*C.L*	*New C.L*
AirAsia	*1*	*80%*	*88.00%*
Malaysia Airlines	*1*	*95%*	*100.00%*
FireFly	*1*	*80%*	*88.00%*
Singapore Airlines	*1*	*85%*	*94.00%*
Service X	*0.5*	*5%*	*53.00%*

❖ ***Discussion***

This scenario is the basis for comparing the features of AMW in scenarios 2 and 3. Every situation of this scenario is normal, and there is no exception in this scenario. Therefore, the results of following sections are compared with the results of this scenario.

6.1.1.2 Scenario 2

This section describes the feature of default weight proposed in AMW. In this scenario, there are no default criteria weights and the user must express the criteria weights. Suppose that the user is not familiar with criteria weighting and selects weights that are not appropriate for the criteria. Similar to scenario 1, scenario 2 comprises three stages.

❖ ***Pre-selection***

This stage is similar to pre-selection in scenario 1 except that the user selects the default weights:

$$Weight_i = \begin{bmatrix} P & A & S & F & R & T & Se & TOM & RS \\ 0.01 & 0.04 & 0.04 & 0.45 & 0.03 & 0.05 & 0.05 & 0.17 & 0.16 \end{bmatrix}$$

However, the rates of services are exactly the same as those in scenario 1, i.e., the rates are collected from the SP and expert DMs with respect to CL and power DMs, respectively. Therefore, the final rates are as shown in Table 6.17.

Table 6.17 Final rates of the alternatives with *TOM* and *RS*

QoS / *Alternative*	P	A	S	F	R	T	Se	TOM	RS
AirAsia	*5.5*	*7*	*5.5*	*2.35*	*6.39*	*5.66*	*6.78*	*5*	*80*
Malaysia Airlines	*8.78*	*7.88*	*8.85*	*6.16*	*8.61*	*9.51*	*8.78*	*5*	*90*
FireFly	*5.88*	*6.48*	*4.76*	*3.28*	*5.31*	*5.66*	*6.39*	*4*	*70*
Singapore Airlines	*7.1*	*7.55*	*7.38*	*8.8*	*8.23*	*8.33*	*9.06*	*5*	*80*
Service X	*5.02*	*4.49*	*5.08*	*3.6*	*4.85*	*5.07*	*5.58*	*3*	*50*

❖ ***Selection***

This stage remains unchanged from scenario 1. Thus, the process is the same as that shown in the flowchart described in Figure 5.4.

Step 1. As criteria weights do not affect the best (f_j^*) and worst (f_j^-) values, these values are the same as those values calculated in scenario 1.

Step 2. Similarly, the normalisation matrix is the same as in scenario 1.

Step 3. The results are changed using the criteria weights for calculatingS_i and R_i. The new values are shown in Table 6.18.

Table 6.18 Values of S_i and R_i

Alternative	S_i	R_i
AirAsia	*0.18*	*0.04*
Malaysia Airlines	*0.27*	*0.27*
FireFly	*0.49*	*0.17*
Singapore Airlines	*0.51*	*0.45*
Service X	*0.64*	*0.17*

Step 4. In this step, S^-, S^+, R^- and R^- are calculated to the specify index values Q_i.

$$S^- = 0.18 \ , \ S^+ = 0.64 \ \text{and} \ R^- = 0.04 \ , \ R^+ = 0.45$$

$$Q_i = \begin{matrix} AirAsia \\ Malaysia\ Airlines \\ FireFly \\ Singapore\ Airlines \\ Service\ X \end{matrix} \begin{bmatrix} 0 \\ 0.378 \\ 0.495 \\ 0.859 \\ 0.659 \end{bmatrix}$$

Step 5. The ranking of three lists are shown in Table 6.19.

Table 6.19 Ranking of the alternatives

Alternative	*Rank of*		
	S_i	R_i	Q_i
AirAsia	*1*	*1*	*1*
Malaysia Airlines	*2*	*3*	*2*
FireFly	*3*	*2*	*3*
Singapore Airlines	*4*	*4*	*5*
Service X	*5*	*2*	*4*

Step 6. The final ranking based on scenario 2,in which the proposed weights by AMW are ignored and the junior user expresses the rates of the alternatives, are as follows:

$$\boldsymbol{AirAsia} > \boldsymbol{MalaysiaAirlines} > \boldsymbol{FireFly} > \boldsymbol{Service\ X} > \boldsymbol{Singapore\ Airlines}$$

- ***Post-selection***

The post-selection stage remains unchanged for this scenario. Therefore, the process of this section follows the stage of post-selection in scenario 1.

❖ ***Discussion***

In this scenario, the feature of default weight provided by AMW is demonstrated. The result is demonstrated to be incorrect because the ranking of services is completely changed. Therefore, if the user is not an expert and is not familiar with the criteria weighting and there is no default weight via AMW, the service consumer must express the weights. In this case, the weights expressed by the user are not accurate, which affects the service selection results. This result is analysed in Sections 6.3 and 6.5.

6.1.1.3 Scenario 3

This scenario demonstrates the service selection results without the CL of the SP. The scenario is the same as scenario 1, except that it does not include the CL of the SP.

❖ ***Pre-selection***

This stage is similar to the pre-selection of scenario 1, except that the rates of the alternatives differ. The aggregated rates of the alternatives without the CL of the SP are shown in Table 6.20.

Table 6.20 Aggregated rates without the CL of the SP

QoS / *Alternative*	P	A	S	F	R	T	Se	TOM	RS
AirAsia	*6.1*	*7.8*	*6.1*	*2.1*	*7.09*	*6.26*	*7.58*	*5*	*80*
Malaysia Airlines	*9*	*8.08*	*9.08*	*6*	*8.84*	*9.76*	*9*	*5*	*90*
FireFly	*6.58*	*7.18*	*5.26*	*2.91*	*5.91*	*6.26*	*7.09*	*4*	*70*
Singapore Airlines	*7.7*	*8.15*	*7.98*	*8.1*	*8.91*	*9*	*9.81*	*5*	*80*
Service X	*7.27*	*6.49*	*7.33*	*2.6*	*7.1*	*7.32*	*8.08*	*4*	*50*

❖ ***Selection***

In this section, the flowchart described in Figure 5.4 is followed. As the rates of alternatives are changed, all of the following steps are affected.

Step 1. The best (f_j^*) and worst (f_j^-) values are shown in Table 6.21.

Table 6.21 Best and worst values for all criteria

Criteria	P	A	S	F	R	T	Se	TOM	RS
f_j^*	*9*	*8.15*	*9.08*	*2.1*	*8.91*	*9.76*	*9.81*	*5*	*90*
f_j^-	*6.1*	*6.49*	*5.26*	*8.1*	*5.91*	*6.26*	*7.09*	*4*	*50*

Step 2. The normalised matrix is shown in Table 6.22.

Table 6.22 Normalised decision matrix

QoS / ***Alternative***	**P**	**A**	**S**	**F**	**R**	**T**	**Se**	**TOM**	**RS**
AirAsia	*1*	*0.21*	*0.78*	*0*	*0.61*	*1*	*0.82*	*0*	*0.25*
Malaysia Airlines	*0*	*0.04*	*0*	*0.65*	*0.02*	*0*	*0.3*	*0*	*0*
FireFly	*0.83*	*0.58*	*1*	*0.14*	*1*	*1*	*1*	*1*	*0.5*
Singapore Airlines	*0.45*	*0*	*0.29*	*1*	*0*	*0.22*	*0*	*0*	*0.25*
Service X	*0.6*	*1*	*0.46*	*0.08*	*0.6*	*0.7*	*0.64*	*1*	*1*

Step 3. S_i and R_i are shown in Table 6.23.

Table 6.23 Values of S_i and R_i

Alternative	S_i	R_i
AirAsia	*0.43*	*0.22*
Malaysia Airlines	*0.12*	*0.09*
FireFly	*0.7*	*0.18*
Singapore Airlines	*0.3*	*0.14*
Service X	*0.72*	*0.17*

Step 4. Q_i is as follow:

$$Q_i = \begin{matrix} AirAsia \\ Malaysia\ Airlines \\ FireFly \\ Singapore\ Airlines \\ Service\ X \end{matrix} \begin{bmatrix} 0.758 \\ 0 \\ 0.829 \\ 0.343 \\ 0.808 \end{bmatrix}$$

Step 5. The three lists, S_i, R_i and Q_i, are ranked in ascending order, as shown in Table 6.24.

Table 6.24 Ranking of the alternatives

Alternative	*Rank of*		
	S_i	R_i	Q_i
AirAsia	*3*	*5*	*3*
Malaysia Airlines	*1*	*1*	*1*
FireFly	*4*	*3*	*5*
Singapore Airlines	*2*	*2*	*2*
Service X	*5*	*4*	*4*

Step 6. In this step, the services are ranked based on Q_i,S_i and R_i. The final ranking list based on scenario 3 is as follows:

$$\boldsymbol{MalaysiaAirlines} > \boldsymbol{Singapore\ Airlines} > \boldsymbol{AirAsia} > \boldsymbol{Service\ X} > \boldsymbol{FireFly}$$

❖ ***Post-selection***

The process of this section is the same as the stage of post-selection in scenario 1.

❖ ***Discussion***

This scenario demonstrates how the lack of a CL affects the service selection results. Without the CL of the SP, the rates are cannot be trusted and the result is not accurate. In this scenario, the alternative, *Service X*, cannot be trusted and could be higher in the service ranking. Because the alternative *Service X* includes high rates expressed by the user, it supposes to be selected as a best candidate. However, the aggregated rates proposed by AMW avoid further inaccurate results. Scenario 5 investigates the case in which there are no aggregated rates. The use of the CL of the SP improves the accuracy of AMW. This result is analysed in Sections 6.3 and 6.5.

6.1.2 Hotel Reservation

This case study is used to test the features of AMW that were not addressed in the first case study. This case study, which concerns a hotel reservation web services, is described. Then, scenarios 4, 5 and 6, which are defined in Table 6.2, are employed. In this case, the user wants to find a service for reserving a hotel in Singapore. The functional requirements for reserving a hotel are as follows: the check-in date is 12 December 2012, the check-out date is 15 December 2012 and the additional criteria are five-star hotels in the Orchard Road location.

Based on those functional requirements, six services match the user's needs. The appropriate service based on the QoS and non-functional properties is selected. In following scenarios, the features of AMW are discussed.

6.1.2.1 Scenario 4

This scenario demonstrates all of the features of AMW in a standard situation for hotel reservation as a basic scenario in normal conditions for comparison with scenarios 5 and 6.

- ***Pre-selection***

In this step, the criteria weights and rates of the alternatives are provided. Therefore, the default weights are applicable for this scenario:

$$Weight_i = \begin{bmatrix} P & A & S & F & R & T & Se & TOM & RS \\ 0.22 & 0.14 & 0.04 & 0.14 & 0.03 & 0.05 & 0.05 & 0.17 & 0.16 \end{bmatrix}$$

In the second step, the rates of these services are collected from the SP. The rates (0-10) from the SP are as follows:

Table 6.25 Rates of the alternatives from the SP

QoS / Alternative	P	A	S	F	R	T	Se
Traders	9	8	7.5	4	8	9	8
Mandarin Orchard	8.5	7	6.5	5.5	7.5	6	7
Pan Pacific	9	7.5	8	5.5	6	7	6
Singapore Marriott	8.5	6	7	6	7	7	6
Sheraton Towers	8	7	6	8.5	9	8	7
Service Y	9.5	8.5	9	3	9	9	10

This rate should be considered by the CL for each service. The CLs for the above services are shown in Table 6.26.

Table 6.26 CLs of the SPs

Hotel Service	CL
Traders	90%
Mandarin Orchard	90%
Pan Pacific	90%
Singapore Marriott	85%
Sheraton Towers	80%
Service Y	25%

In Table 6.27, the rates are shown with respect to the CLs.

Table 6.27 Rates of the alternatives from SPs with CLs

QoS / Alternative	P	A	S	F	R	T	Se
Traders	4.8	6.4	4.8	2.5	5.6	4.8	6.4
Mandarin Orchard	8.55	7.6	8.55	6.32	8.55	9.5	8.55
Pan Pacific	5.6	5.6	4	3.75	4.8	4.8	5.6
Singapore Marriott	6.8	6.8	6.8	9.41	7.65	7.65	8.5
Sheraton Towers	4.5	4	4.5	4	4.5	4.5	5
Service Y	4.5	4	4.5	4	4.5	4.5	5

In this scenario, the rates are collected from expert DMs using linguistic data. It is occasionally easier for the expert DMs to specify the ratings in terms of linguistic information instead of numerical data. In this scenario, the applicability of the ratings is discussed. For this purpose, [89] proposes a mapping table that consists of 11 points and therefore allows users to represent their preferences easily.

Table 6.28 Values of the service selection factors in the 11-point scale format

Linguistic variables	*Assigned value*
Exceptionally low (XL)	*0.5*
Extremely low (EL)	*1.5*
Very low (VL)	*2.5*
Low (L)	*3.5*
Below average (BA)	*4.0*
Average (A)	*5.0*
Above average (AA)	*6.0*
High (H)	*6.5*
Very high (VH)	*7.5*
Extremely high (EH)	*8.5*
Exceptionally high (XH)	*9.5*

The rates of three expert DMs in linguistic format are shown in Table 6.29.

Table 6.29 Linguistic rates of the alternatives from the expert DMs

	QoS / ***Alternative***	***P***	***A***	***S***	***F***	***R***	***T***	***Se***
d_1	***Traders***	*EH*	*H*	*H*	*BA*	*VH*	*EH*	*VH*
	Mandarin Orchard	*EH*	*H*	*AA*	*AA*	*H*	*A*	*H*
	Pan Pacific	*XH*	*H*	*VH*	*AA*	*A*	*H*	*AA*
	Singapore Marriott	*EH*	*A*	*AA*	*VH*	*AA*	*AA*	*A*
	Sheraton Towers	*VH*	*AA*	*AA*	*EH*	*VH*	*H*	*A*
	Service Y	*VL*	*L*	*VL*	*XH*	*VL*	*VL*	*EL*
d_2	***Traders***	*EH*	*VH*	*AA*	*BA*	*H*	*VH*	*H*
	Mandarin Orchard	*VH*	*H*	*AA*	*A*	*VH*	*AA*	*AA*
	Pan Pacific	*EH*	*H*	*H*	*AA*	*AA*	*H*	*A*
	Singapore Marriott	*VH*	*A*	*AA*	*H*	*AA*	*AA*	*A*
	Sheraton Towers	*H*	*A*	*A*	*XH*	*VH*	*H*	*AA*
	Service Y	*VL*	*VL*	*EL*	*XH*	*VL*	*EL*	*VL*
d_3	***Traders***	*EH*	*VH*	*VH*	*A*	*VH*	*EH*	*VH*
	Mandarin Orchard	*VH*	*AA*	*A*	*AA*	*H*	*A*	*H*
	Pan Pacific	*VH*	*H*	*VH*	*A*	*A*	*H*	*A*
	Singapore Marriott	*VH*	*A*	*AA*	*VH*	*AA*	*AA*	*A*
	Sheraton Towers	*AA*	*AA*	*A*	*EH*	*H*	*H*	*AA*
	Service Y	*L*	*VL*	*VL*	*XH*	*EL*	*VL*	*L*

The powers of the DMs, from which the rates of alternatives will be aggregated, are as follows:

Table 6.30 Power of the expert DMs

Decision Maker	*CL*
d1	*35%*
d2	*32%*
d3	*33%*

The numeric rates from the DMs with respect to the power of the DMs are aggregated, as shown below.

Table 6.31 Numeric aggregated rates of the alternatives from the expert DMs

QoS / *Alternative*	P	A	S	F	R	T	Se
Traders	*8.5*	*7.15*	*6.67*	*4.33*	*7.18*	*8.18*	*7.18*
Mandarin Orchard	*7.85*	*6.34*	*5.67*	*5.68*	*6.82*	*5.32*	*6.34*
Pan Pacific	*8.52*	*6.5*	*7.18*	*5.67*	*5.32*	*6.5*	*5.35*
Singapore Marriott	*7.85*	*5*	*6*	*7.18*	*6*	*6*	*5*
Sheraton Towers	*6.69*	*5.68*	*5.35*	*8.82*	*7.17*	*6.5*	*5.65*
Service Y	*2.83*	*2.85*	*2.18*	*9.5*	*2.17*	*2.18*	*2.48*

The final average rates from the SP and expert DMs with respect to Algorithm 2 are shown below.

Table 6.32 Final average rates of the alternatives

QoS / *Alternative*	P	A	S	F	R	T	Se
Traders	*8.3*	*7.18*	*6.71*	*4.39*	*7.19*	*8.14*	*7.19*
Mandarin Orchard	*7.75*	*6.32*	*5.76*	*5.9*	*6.79*	*5.36*	*6.32*
Pan Pacific	*8.31*	*6.63*	*7.19*	*5.89*	*5.36*	*6.4*	*5.38*
Singapore Marriott	*7.54*	*5.05*	*5.98*	*7.12*	*5.98*	*5.98*	*5.05*
Sheraton Towers	*6.55*	*5.64*	*5.08*	*9.73*	*7.19*	*6.45*	*5.63*
Service Y	*2.6*	*2.49*	*2.22*	*10.75*	*2.21*	*2.22*	*2.49*

By adding two additional criteria, *TOM* and *RS,* the final decision matrix is completed.

Table 6.33 Final rates of the alternatives with *TOM* and *RS*

QoS / *Alternative*	P	A	S	F	R	T	Se	TOM	RS
Traders	*8.3*	*7.18*	*6.71*	*4.39*	*7.19*	*8.14*	*7.19*	*5*	*90*
Mandarin Orchard	*7.75*	*6.32*	*5.76*	*5.9*	*6.79*	*5.36*	*6.32*	*5*	*80*
Pan Pacific	*8.31*	*6.63*	*7.19*	*5.89*	*5.36*	*6.4*	*5.38*	*4*	*70*
Singapore Marriott	*7.54*	*5.05*	*5.98*	*7.12*	*5.98*	*5.98*	*5.05*	*4*	*85*
Sheraton Towers	*6.55*	*5.64*	*5.08*	*9.73*	*7.19*	*6.45*	*5.63*	*4*	*70*
Service Y	*2.6*	*2.49*	*2.22*	*10.75*	*2.21*	*2.22*	*2.49*	*3*	*75*

❖ ***Selection***

The decision matrix provided in the previous section is given to Algorithm 3 of AMW to select the appropriate service based on the data provided in the prior section.

Step 1. The best (f_j^*) and worst (f_j^-) values for each column are located, which are shown below (financial is a negative criterion):

Table 6.34 Best and worst values for all criteria

Criteria	P	A	S	F	R	T	Se	TOM	RS
f_j^*	*8.31*	*7.18*	*7.19*	*4.39*	*7.19*	*8.14*	*7.19*	*5*	*90*
f_j^-	*2.6*	*2.49*	*2.22*	*10.75*	*2.21*	*2.22*	*2.49*	*3*	*70*

Step 2. The decision matrix must be normalised by applying linear normalisation. The normalised matrix is shown below.

Table 6.35 Normalised decision matrix

QoS / *Alternative*	P	A	S	F	R	T	Se	TOM	RS
Traders	*0*	*0*	*0.1*	*0*	*0*	*0*	*0*	*0*	*0*
Mandarin Orchard	*0.1*	*0.18*	*0.29*	*0.24*	*0.08*	*0.47*	*0.19*	*0*	*0.5*
Pan Pacific	*0*	*0.12*	*0*	*0.24*	*0.37*	*0.29*	*0.39*	*0.5*	*1*
Singapore Marriott	*0.13*	*0.45*	*0.24*	*0.43*	*0.24*	*0.36*	*0.46*	*0.5*	*0.25*
Sheraton Towers	*0.31*	*0.33*	*0.42*	*0.84*	*0*	*0.29*	*0.33*	*0.5*	*1*
Service Y	*1*	*1*	*1*	*1*	*1*	*1*	*1*	*1*	*0.75*

Step 3. Based on the normalised matrix, the appropriate matrix used to compare S_i and R_i can be determined.

Table 6.36 Values of S_i and R_i

Alternative	S_i	R_i
Traders	*0*	*0*
Mandarin Orchard	*0.2*	*0.08*
Pan Pacific	*0.34*	*0.16*
Singapore Marriott	*0.34*	*0.09*
Sheraton Towers	*0.54*	*0.16*
Service Y	*0.96*	*0.22*

Step 4. In this step, S^-, S^+, R^- and $R^-$are calculated to specify the index values Q_i:

$$S^- = 0.0 \ , \ S^+ = 0.96 \text{ and } R^- = 0.0 \ , \ R^+ = 0.22$$

$$Q_i = \begin{matrix} Traders \\ Mandarin\ Orchard \\ Pan\ Pacific \\ Singapore\ Marriott \\ Sheraton\ Towers \\ Service\ Y \end{matrix} \begin{bmatrix} 0 \\ 0.286 \\ 0.541 \\ 0.382 \\ 0.645 \\ 1 \end{bmatrix}$$

Step 5. The ranking of the three lists are as follows:

Table 6.37 Ranking of the alternatives

Alternative	*Rank of* S_i	R_i	Q_i
Traders	*1*	*1*	*1*
Mandarin Orchard	*2*	*2*	*2*
Pan Pacific	*3*	*4*	*4*
Singapore Marriott	*3*	*3*	*3*
Sheraton Towers	*4*	*5*	*5*
Service Y	*5*	*6*	*6*

Step 6. The service for *Traders Hotel* is the best option with respect to the QoS criteria, whereas *Service Y* is the worst option. The final ranking list obtained in scenario 4 is as follows:

$$\boldsymbol{Traders > Mandarin\ Orchard > Singapore\ Marriott > Pan\ Pacific > Sheraton\ Towers > Service\ Y}$$

❖ ***Post-selection***

The CL is estimated based on Algorithm 4. First, the similarity of the data are calculated by comparing the data of the SP and expert DMs:

Table 6.38 Calculated similarity rates

QoS / ***Alternative***	**P**	**A**	**S**	**F**	**R**	**T**	**Se**
Traders	*1*	*1*	*1*	*1*	*1*	*1*	*1*
Mandarin Orchard	*1*	*1*	*1*	*1*	*1*	*1*	*1*
Pan Pacific	*1*	*1*	*1*	*1*	*1*	*1*	*1*
Singapore Marriott	*1*	*1*	*1*	*1*	*1*	*1*	*1*
Sheraton Towers	*1*	*1*	*1*	*1*	*1*	*1*	*1*
Service Y	*-1*	*0.5*	*-1*	*-1*	*-1*	*-1*	*-1*

Then, the average similarity and updated CLs of the SPs are estimated:

Table 6.39 Estimated CLs

Alternative	*Average*	*C.L*	*New C.L*
Traders	*1*	*90%*	*99.00%*
Mandarin Orchard	*1*	*90%*	*99.00%*
Pan Pacific	*1*	*90%*	*99.00%*
Singapore Marriott	*1*	*85%*	*94.00%*
Sheraton Towers	*1*	*80%*	*88.00%*
Service Y	*-0.79*	*25%*	*23.00%*

❖ ***Discussion***

This scenario is the basis for comparing the features of AMW in scenarios 5 and 6. Every situation of this scenario is normal, and there is no exception in this scenario. The results of the following sections are compared with the results of this scenario.

6.1.2.2 Scenario 5

In this scenario, the rates of the SP are used directly without being aggregated with the rates of the expert DMs.

❖ ***Pre-selection***

The pre-selection stage of scenario 5 is the same as that of scenario 4, except the rates are different.

Table 6.40 Rates of the alternatives from the SP

QoS / ***Alternative***	***P***	***A***	***S***	***F***	***R***	***T***	***Se***
Traders	*9*	*8*	*7.5*	*4*	*8*	*9*	*8*
Mandarin Orchard	*8.5*	*7*	*6.5*	*5.5*	*7.5*	*6*	*7*
Pan Pacific	*9*	*7.5*	*8*	*5.5*	*6*	*7*	*6*
Singapore Marriott	*8.5*	*6*	*7*	*6*	*7*	*7*	*6*
Sheraton Towers	*8*	*7*	*6*	*8.5*	*9*	*8*	*7*
Service Y	*9.5*	*8.5*	*9*	*3*	*9*	*9*	*10*

The above rates are used in addition to the rates of two additional criteria, *TOM* and *RS*:

Table 6.41 Final rates of alternatives with *TOM* and *RS*

QoS / *Alternative*	P	A	S	F	R	T	Se	TOM	RS
Traders	9	8	7.5	4	8	9	8	5	90
Mandarin Orchard	8.5	7	6.5	5.5	7.5	6	7	4	80
Pan Pacific	9	7.5	8	5.5	6	7	6	5	70
Singapore Marriott	8.5	6	7	6	7	7	6	5	85
Sheraton Towers	8	7	6	8.5	9	8	7	4	70
Service Y	9.5	8.5	9	3	9	9	10	3	75

- ***Selection***

Step 1. The best (f_j^*) and worst (f_j^-) values for each column are located, as shown below (financial is a negative criterion).

Table 6.42 Best and worst values for all criteria

Criteria	P	A	S	F	R	T	Se	TOM	RS
f_j^*	9.5	8.5	9	3	9	9	10	5	90
f_j^-	8	6	6	8.5	6	6	6	3	70

Step 2. The normalised matrix is shown in below:

Table 6.43 Normalised decision matrix

QoS / *Alternative*	P	A	S	F	R	T	Se	TOM	RS
Traders	0.33	0.2	0.5	0.18	0.33	0	0.5	0	0
Mandarin Orchard	0.67	0.6	0.83	0.45	0.5	1	0.75	0	0.5
Pan Pacific	0.33	0.4	0.33	0.45	1	0.67	1	0.5	1
Singapore Marriott	0.67	1	0.67	0.55	0.67	0.67	1	0.5	0.25
Sheraton Towers	1	0.6	1	1	0	0.33	0.75	0.5	1
Service Y	0	0	0	0	0	0	0	1	0.75

Step 3. Based on the normalised matrix, the appropriate matrix used to compare S_i and R_i can be determined.

Table 6.44 Values of S_i and R_i

Alternative	S_i	R_i
Traders	*0.19*	*0.07*
Mandarin Orchard	*0.51*	*0.15*
Pan Pacific	*0.56*	*0.16*
Singapore Marriott	*0.63*	*0.15*
Sheraton Towers	*0.79*	*0.22*
Service Y	*0.29*	*0.17*

Step 4. In this step, S^-, S^+, R^- and $R^-$are calculated in to specify the index values Q_i:

$$S^- = 0.19\ ,\ S^+ = 0.79 \text{ and } R^- = 0.07\ ,\ R^+ = 0.22$$

$$Q_i = \begin{matrix} Traders \\ Mandarin\ Orchard \\ Pan\ Pacific \\ Singapore\ Marriott \\ Sheraton\ Towers \\ Service\ Y \end{matrix} \begin{bmatrix} 0 \\ 0.533 \\ 0.608 \\ 0.633 \\ 1 \\ 0.417 \end{bmatrix}$$

Step 5. The ranking the three lists is shown below.

Table 6.45 Ranking of the alternatives

Alternative	*Rank of*		
	S_i	R_i	Q_i
Traders	*1*	*1*	*1*
Mandarin Orchard	*3*	*2*	*3*
Pan Pacific	*4*	*3*	*4*
Singapore Marriott	*5*	*2*	*5*
Sheraton Towers	*6*	*5*	*6*
Service Y	*2*	*4*	*2*

Step 6. The final ranking list obtained in scenario 5 is as follows:

$$\boldsymbol{Traders > Service\ Y > Mandarin\ Orchard > Pan\ Pacific > Singapore\ Marriott > Sheraton\ Towers}$$

❖ ***Post-selection***

As the rates are collected from the SP directly and there are no DM rates for comparison, this stage is not applicable for this scenario.

❖ ***Discussion***

The result of this scenario demonstrates that the ranking is inaccurate because the rates are collected from SPs directly and are not aggregated with the rates from DMs. As the SP of *Service Y* cannot be trusted, the expressed rates are not accurate. The selection result is affected by inaccurate data. As shown in the ranking result in step 6, the location of *Service Y* is ranked second instead of being ranked last. Therefore, simply trusting the SP significantly affected the service selection results. Furthermore, the accuracy of AMW is improved by aggregating the rates of the SP with the rates of expert DMs. This result is analysed in Sections 6.3 and 6.5.

6.1.2.3 Scenario 6

The aim of this scenario is to demonstrate the features of AMW that were not covered by prior scenarios, specifically, *D2* and *E2* in Table 6.1. *D2* is the case in which all alternatives have the same values for a criterion. *E2* is the case in which the maturity of criteria is not addressed. This scenario demonstrates how AMW avoids any issue in the service selection process. In this scenario, the values of *trust* are the same for all alternatives. Furthermore, one of the alternatives does not address the maturity of criteria.

❖ ***Pre-selection***

Although the default weights should be changed later, the default weights are as follows:

$$Weight_i = \begin{bmatrix} P & A & S & F & R & T & Se & TOM & RS \\ 0.22 & 0.14 & 0.04 & 0.14 & 0.03 & 0.05 & 0.05 & 0.17 & 0.16 \end{bmatrix}$$

The aggregated rates of the alternatives are as follows:

Table 6.46 Aggregated rates of alternatives with *TOM* and *RS*

QoS / *Alternative*	P	A	S	F	R	T	Se	TOM	RS
Traders	*9.5*	*0.76*	*1.02*	*1.01*	*0.77*	*4*	*1*	*5*	*90*
Mandarin Orchard	*8.18*	*6.67*	*6.09*	*5.59*	*7.16*	*4*	*6.67*	*3*	*80*
Pan Pacific	*8.76*	*7*	*7.59*	*5.59*	*5.66*	*4*	*5.68*	*4*	*70*
Singapore Marriott	*8.18*	*5.5*	*6.5*	*6.59*	*6.5*	*4*	*5.5*	*3*	*85*
Sheraton Towers	*7.35*	*6.34*	*5.68*	*8.66*	*8.09*	*4*	*6.33*	*4*	*70*
Service Y	*6.17*	*5.68*	*5.59*	*6.25*	*5.59*	*4*	*6.24*	*3*	*75*

❖ ***Selection***

Based on the AMW approach, in the above situation, the similar rates and their criterion are eliminated from the decision matrix, i.e., the rates for *trust* are deleted. Therefore, the decision matrix is revised by the process specified in the first part of Algorithm 3. The final aggregated decision matrix is shown below.

Table 6.47 Final rates of the alternatives with *TOM* and *RS*

QoS / *Alternative*	P	A	S	F	R	Se	TOM	RS
Traders	*9.5*	*0.76*	*1.02*	*1.01*	*0.77*	*1*	*5*	*90*
Mandarin Orchard	*8.18*	*6.67*	*6.09*	*5.59*	*7.16*	*6.67*	*3*	*80*
Pan Pacific	*8.76*	*7*	*7.59*	*5.59*	*5.66*	*5.68*	*4*	*70*
Singapore Marriott	*8.18*	*5.5*	*6.5*	*6.59*	*6.5*	*5.5*	*3*	*85*
Sheraton Towers	*7.35*	*6.34*	*5.68*	*8.66*	*8.09*	*6.33*	*4*	*70*
Service Y	*6.17*	*5.68*	*5.59*	*6.25*	*5.59*	*6.24*	*3*	*75*

As the number of criteria is reduced, the default weights should be revised simultaneously. Therefore, the default weights are calculated again via Algorithm 1. The revised default weights are as follows:

$$Weight_i = \begin{bmatrix} P & A & S & F & R & Se & TOM & RS \\ 0.22 & 0.14 & 0.04 & 0.14 & 0.05 & 0.05 & 0.18 & 0.18 \end{bmatrix}$$

Finally, AMW selects the best service with eight criteria and six service alternatives.

Step 1. The best (f_j^*) and worst (f_j^-) values for each column are located, as shown below (financial is a negative criterion).

Table 6.48 Best and worst values for all criteria

Criteria	**P**	**A**	**S**	**F**	**R**	**Se**	**TOM**	**RS**
f_j^*	*9.5*	*7*	*7.59*	*1.01*	*8.09*	*6.67*	*5*	*90*
f_j^-	*6.17*	*0.76*	*1.02*	*8.66*	*0.77*	*1*	*3*	*70*

Step 2. The normalised matrix is shown in below:

Table 6.49 Normalised decision matrix

QoS / ***Alternative***	**P**	**A**	**S**	**F**	**R**	**Se**	**TOM**	**RS**
Traders	*0*	*1*	*1*	*0*	*1*	*1*	*0*	*0*
Mandarin Orchard	*0.4*	*0.05*	*0.23*	*0.6*	*0.13*	*0*	*1*	*0.5*
Pan Pacific	*0.22*	*0*	*0*	*0.6*	*0.33*	*0.17*	*0.5*	*1*
Singapore Marriott	*0.4*	*0.24*	*0.17*	*0.73*	*0.22*	*0.21*	*1*	*0.25*
Sheraton Towers	*0.65*	*0.11*	*0.29*	*1*	*0*	*0.06*	*0.5*	*1*
Service Y	*1*	*0.21*	*0.3*	*0.68*	*0.34*	*0.08*	*1*	*0.75*

Step 3. The appropriate matrix used to compare S_i and R_i can be determined.

Table 6.50 Values of S_i and R_i

Alternative	S_i	R_i
Traders	*0.28*	*0.14*
Mandarin Orchard	*0.47*	*0.18*
Pan Pacific	*0.43*	*0.18*
Singapore Marriott	*0.48*	*0.18*
Sheraton Towers	*0.58*	*0.18*
Service Y	*0.7*	*0.22*

Step 4. In this step, S^{-}, S^{+}, $R^{-}$and $R^{-}$are calculated to specify the index values Q_i:

$$S^{-} = 0.28, S^{+} = 0.7 \text{ and } R^{-} = 0.14, R^{+} = 0.22$$

$$Q_i = \begin{matrix} Traders \\ Mandarin\ Orchard \\ Pan\ Pacific \\ Singapore\ Marriott \\ Sheraton\ Towers \\ Service\ Y \end{matrix} \begin{bmatrix} 0 \\ 0.476 \\ 0.429 \\ 0.488 \\ 0.607 \\ 1 \end{bmatrix}$$

Step 5. The ranking of the three lists is shown below.

Table 6.51 Ranking of the alternatives

Alternative	*Rank of*		
	S_i	R_i	Q_i
Traders	*1*	*1*	*1*
Mandarin Orchard	*3*	*2*	*3*
Pan Pacific	*2*	*2*	*2*
Singapore Marriott	*4*	*2*	*4*
Sheraton Towers	*5*	*2*	*5*
Service Y	*6*	*3*	*6*

Step 6. The final ranking list for scenario 6 is as follows:

Traders > ***Pan Pacific*** > ***Mandarin Orchard*** > ***Singapore Marriott*** > ***Sheraton Towers*** > ***Service Y***

Although the ranking is finished, the selection stage is still not complete. Considering the decision matrix, the scores of only four criteria in the *Traders* service are higher than the other alternatives. Although the *Traders* service has the highest rates in *performance, financial, TOM* and *RS,* it has the lowest rates in the remaining criteria. Therefore, the maturity of criteria is not addressed, and based on Algorithm 3,*Pan Pacific*, which has the second-highest ranking, should be selected.

❖ ***Discussion***

As mentioned earlier, this scenario attempts to demonstrate the features of AMW that were not covered by prior scenarios. The situations of neglecting the maturity of criteria and the similarity of rates for one criterion are covered in this scenario. The results demonstrate how AMW avoids any problem in the service selection process. This result is analysed in Sections 6.3 and 6.5.

6.2 Results of the Existing Approach

In this section, the proposed service selection approach via WSMO is applied on the two case studies described in Section 6.1. The aim of applying AMW on the case studies is to present the features of AMW, whereas the aim of this section is to validate those features. The author attempts to show what would be the results without those features.

This section identifies the types of problems that AMW can solve that cannot be solved by the existing approaches. These claims are supported by demonstrations of each approach applied to specific examples that illustrate each condition identified in the analysis. To compare and validate the results, the closest approach to AMW, WSMO (described in Section 2.5.4.1), is used. Then, in Section 6.3 and 6.5, the results of the existing and proposed approach are compared.

Although the WSMO algorithm supports multi-criteria, it is a naive algorithm. This approach does not use any proven multi-criteria algorithms, such as AHP, ANP and TOPSIS. The algorithm includes two simple steps: normalisation and ranking.

6.2.1 Flight Booking

As WSMO doesn't support any feature of AMW such as confidence level of service provider, aggregated rates with respect to expert DMs, therefore in all scenario it simply accept the input data from service provider. Therefore the results in all situations are same, so in this case the scenarios are not applicable.

The importance term in WSMO is actually the criteria weights, so the importance is as follows:

$$imp = \begin{bmatrix} P & A & S & F & R & T & Se & TOM & RS \\ 0.22 & 0.14 & 0.04 & 0.14 & 0.03 & 0.05 & 0.05 & 0.17 & 0.16 \end{bmatrix}$$

Furthermore, the non-functional value term is the rates of alternatives. Therefore, the final rates are shown below:

Table 6.52 Non-functional values of the alternatives

QoS / *Alternative*	P	A	S	F	R	T	Se	TOM	RS
AirAsia	6	8	6	2	7	6	8	5	80
Malaysia Airlines	9	8	9	6	9	10	9	5	90
FireFly	7	7	5	3	6	6	7	4	70
Singapore Airlines	8	8	8	8	9	9	10	5	80
Service X	9	8	9	2	9	9	10	3	50

The ranking by WSMO for all scenarios of flight booking is as follow:

$$Malaysia\ Airlines > Service\ X > AirAsia \approx Singapore\ Airlines > FireFly$$

6.2.2 Hotel Reservation

Also in this case study the described scenarios are not applicable and in all states, the results are same.

The importance value is that default weights:

$$imp = \begin{bmatrix} P & A & S & F & R & T & Se & TOM & RS \\ 0.22 & 0.14 & 0.04 & 0.14 & 0.03 & 0.05 & 0.05 & 0.17 & 0.16 \end{bmatrix}$$

Also the non-functional values are:

Table 6.53 Non-functional values of the alternatives

QoS / Alternative	P	A	S	F	R	T	Se	TOM	RS
Traders	*9*	*8*	*7.5*	*4*	*8*	*9*	*8*	*5*	*90*
Mandarin Orchard	*8.5*	*7*	*6.5*	*5.5*	*7.5*	*6*	*7*	*5*	*80*
Pan Pacific	*9*	*7.5*	*8*	*5.5*	*6*	*7*	*6*	*4*	*70*
Singapore Marriott	*8.5*	*6*	*7*	*6*	*7*	*7*	*6*	*4*	*85*
Sheraton Towers	*8*	*7*	*6*	*8.5*	*9*	*8*	*7*	*4*	*70*
Service Y	*9.5*	*8.5*	*9*	*3*	*9*	*9*	*10*	*3*	*75*

The ranking by WSMO for all scenarios of hotel reservation is as follow:

$$\textbf{\textit{Traders}} > \textbf{\textit{Service Y}} > \textbf{\textit{Mandarin Orchard}} > \textbf{\textit{Pan Pacific}} > \textbf{\textit{Singapore Marriott}} > \textbf{\textit{Sheraton Towers}}$$

❖ ***Discussion***

WSMO uses the rate from SP but does not consider the SP's CL. Also WSMO uses the rate from the SP directly without considering the expert DMs. Typically the SP cannot always be trusted, and the rates must be verified by experts. Furthermore WSMO does not use any proven MCDM algorithms; it might rank the services incorrectly, even in the same situation. Therefore, based on the above facts in different scenarios the result of WSMO is same and inaccurate.

6.3 Analysis Result of Case Studies

In this section the result of case studies, described in Section 6.1 and 6.2, are evaluated based on the accuracy measurement that discussed in Chapter 4. For this purpose, the precision-recall graph in terms of the interpolated precision and 11 recall levels is first provided, and then the averaged 11-point interpolated precision-recall graph is depicted. These graphs are repeated for each case study, and finally, a graph of the average interpolated precision and 11 recall levels is depicted. This analysis is replicated for the existing approach, and the results are shown separately. The results of the proposed and existing approach are presented in one graph.

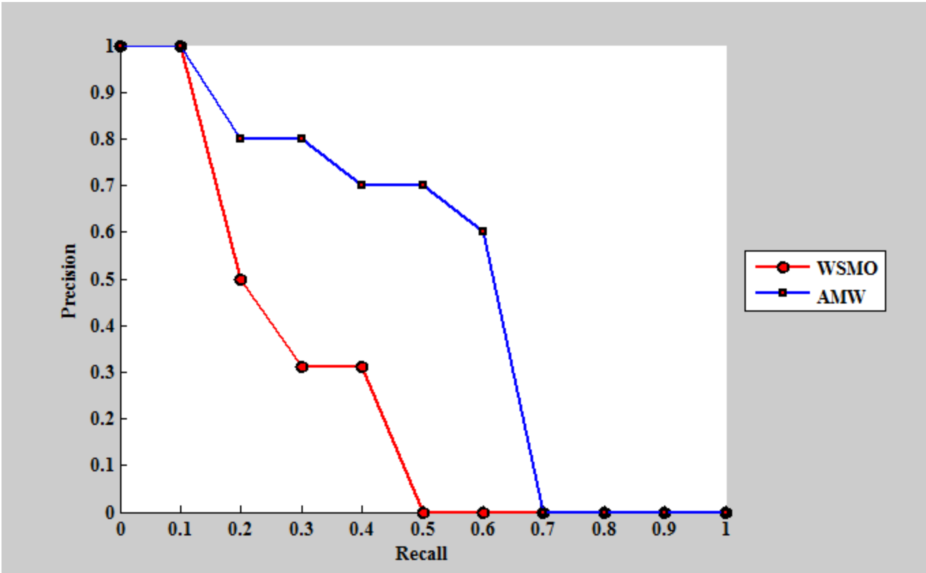

Figure 6.2 Averaged 11-point precision-recall graph across 6 scenarios applied

As described in Chapter 4, the curves closest to the upper right-hand corner of the graph indicate the best accuracy. Comparisons are best made in three different recall ranges: 0 to 0.2, 0.2 to 0.8, and 0.8 to 1. These ranges characterize high precision, middle recall, and high recall, respectively [84]. The curves of the graphs demonstrate that AMW and WSMO are both applicable for WSS. In the above graph the curves demonstrate that the almost values in the high precision and high recall ranges in both approaches are same. But in the middle recall range the curve of AMW is closer to the ideal point. However, the area under the curve (AUC) for AMW is greater than that for WSMO. Therefore, the AMW approach in terms of accuracy outperforms WSMO.

Moreover, as previously mentioned, examining the entire precision-recall curve is very informative, but it is often desirable to summarise this information with a few numbers, or perhaps even a single number. In recent years, other measures have become more common. Most standard among the TREC community is the Mean Average Precision (MAP), which provides a single-figure measure of quality across recall levels. Among the existing evaluation measures, MAP has been shown to have especially good discrimination and stability. The formula for MAP is shown below:

$$\mathrm{MAP}(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} \mathrm{Precision}(R_{jk})$$

In fact, MAP is the average precision value obtained after preparing the average interpolated precision and 11 recall-level graph. Based on this derivation, the MAP values calculated based on above formula are 0.5091 for AMW and 0.2836 for WSMO.

6.4 Experimental Result

To evaluate the proposed approach, the experiment is performed on AMW_TC1. The OWL-S test collection (accessible via http://projects.semwebcentral.org/frs/?group_id=89&release_id=380) is converted into a WSML file, then the non-functional properties of each service are added to the services. Of the 1,008 included web services, 197 are travelling services; the category of travelling services is chosen for the purpose of this research. The goals, which are represented in terms of the WSML file, are provided as the input for the prototype. Because AMW proposed the default criteria weights, the non-functional properties are added to WSML files automatically with respect to those weights. Because the user might want to modify the default weights, the weights are changed randomly every 10 services. All test collections are performed on a computer with the Windows Vista Ultimate Edition 2006 operating systems, an Intel Core 2 Duo 3.00 GHz processor and 2 GB of RAM using JDK 1.6.0.

As discussed, the AMW test collection involves six queries. In this part, each query is described, and the number of services and the number of discovered services are expressed. Then, the 11-point, interpolated, precision-recall graph is depicted. As the last result after the six queries, the averaged 11-point precision-recall graph across all six queries is illustrated.

All graphs in this section are depicted in MATLAB R2012b using an adopted function for ranked retrieval programmed by the author to draw the graph based on the concepts presented in Section 4.2.4.1.

- ***Query 1:***

In this query, “the client aims to know about the destination for surfing”. There are 35 services for this query, but 21 services are discovered. For the purpose of selection, the discovered services should be ranked based on their non-functional properties. AMW successfully ranked 17 services correctly in the ranked list. For this query, the graph of precision-recall is based on the concepts described in Section 4.2.4.1, which is about the precision-recall graph for ranked retrievals. The 11-point interpolated precision-recall graph is shown below.

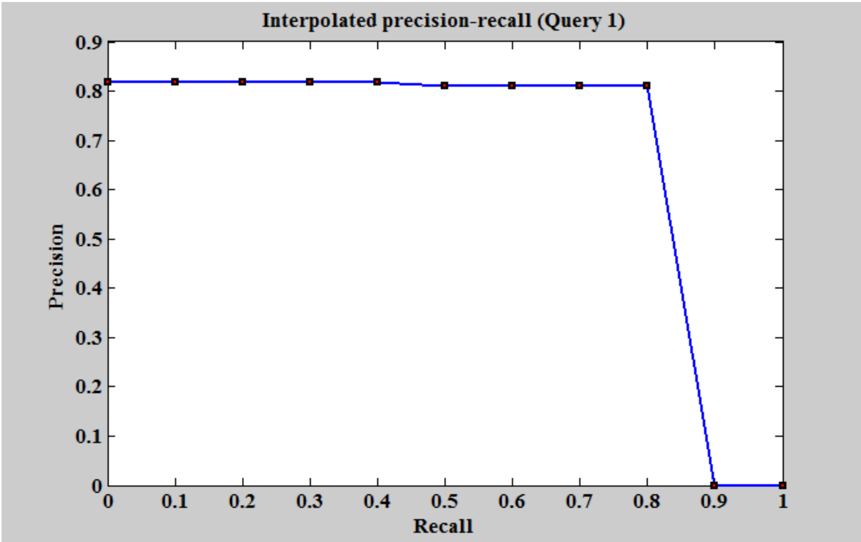

Figure 6.3 The 11-point interpolated precision-recall graph for query 1

❖ ***Query 2:***

There are 41 services for this query, and36 services are discovered. In this query, “the client aims to know about the destination where facilities for sports hiking and surfing available”. The discovered services are ranked using AMW, and 32 of the web services are ranked correctly. Among the discovered web services, the expressed rates of three of the web services are not reliable. However, the confidence level feature and the expert DMs feature of AMW avoid any effect on the ranking results. The 11-point interpolated precision-recall graph is shown below:

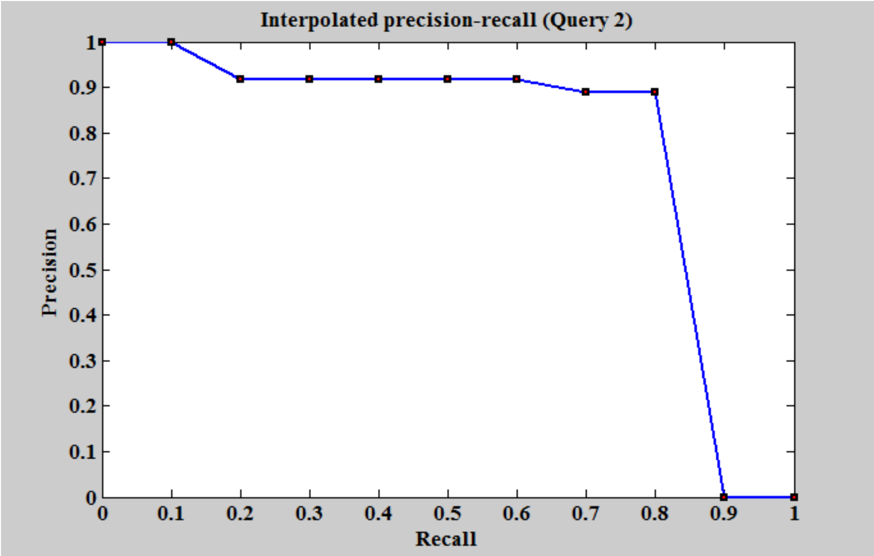

Figure 6.4 The 11-point interpolated precision-recall graph for query 2

- ***Query 3:***

In this query, which includes 15 web services, "the client wants to travel from Frankfurt to Berlin, that's why it puts a request to find a map to locate a route from Frankfurt to Berlin". Based on this query, 11 web services are discovered, of which two are not reliable. However, the graph below illustrates that the curve is close to the ideal point, which is at the top right corner, and incorrect input data could not affect the ranking results.

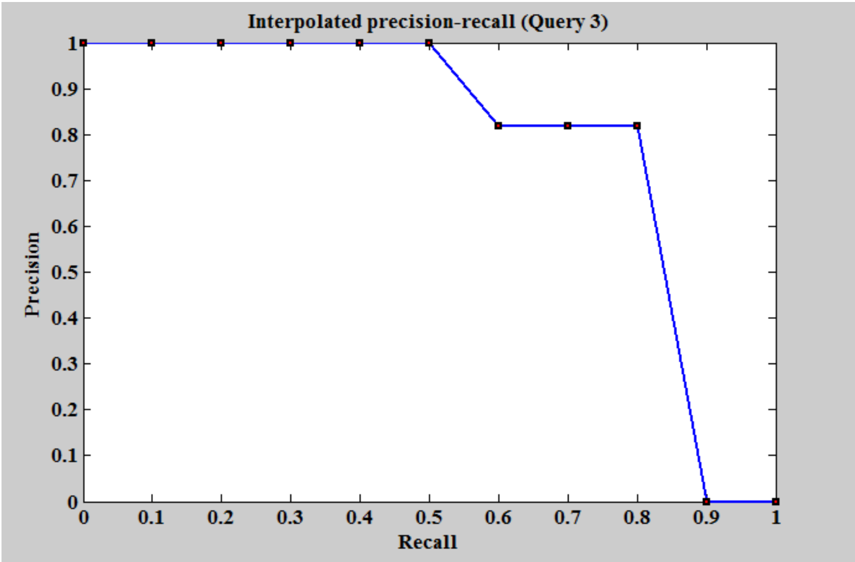

Figure 6.5 The 11-point interpolated precision-recall graph for query 3

❖ ***Query 4:***

This query discovers 11 web services, of which two have unreliable input data. In this query, "the client aims to know about the destination of the organization for a certain type of surfing". Although approximately 20% of the input data are incorrect, AMW improve the accuracy of input data followed by the accuracy of the result. Moreover, the confidence levels of the service providers of those two services are decreased. These confidence levels are considered during the next instance of ranking and selecting services from those two service providers. The 11-point interpolated precision-recall graph resulting from this query is depicted.

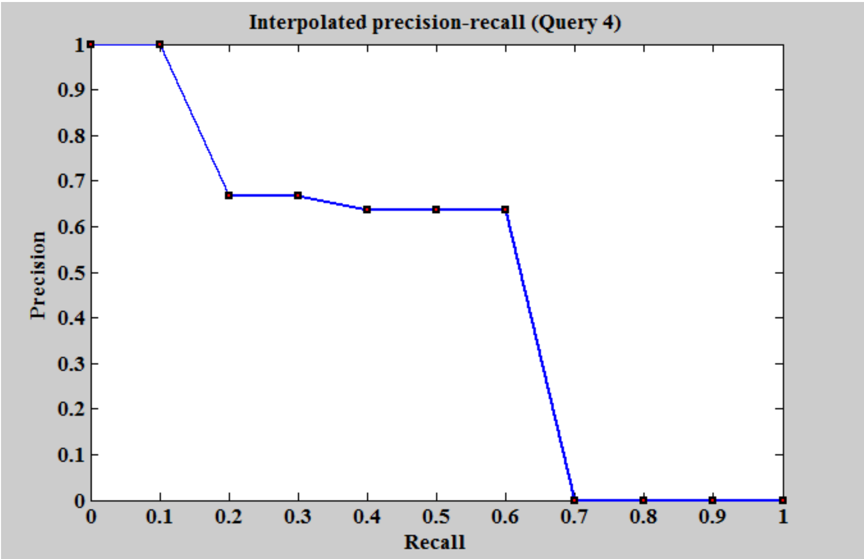

Figure 6.6 The 11-point interpolated precision-recall graph for query 4

❖ ***Query 5:***

In this query, "the client aims to know about hotel in a city of a country". Eleven of the 23 available services for this query are discovered. Additionally, in this query, there are two unreliable web services with the potential to affect the ranking results. However, the accuracy of the result illustrated below by the 11-point interpolated precision-recall graph proves that AMW also avoids an inaccurate result in this situation.

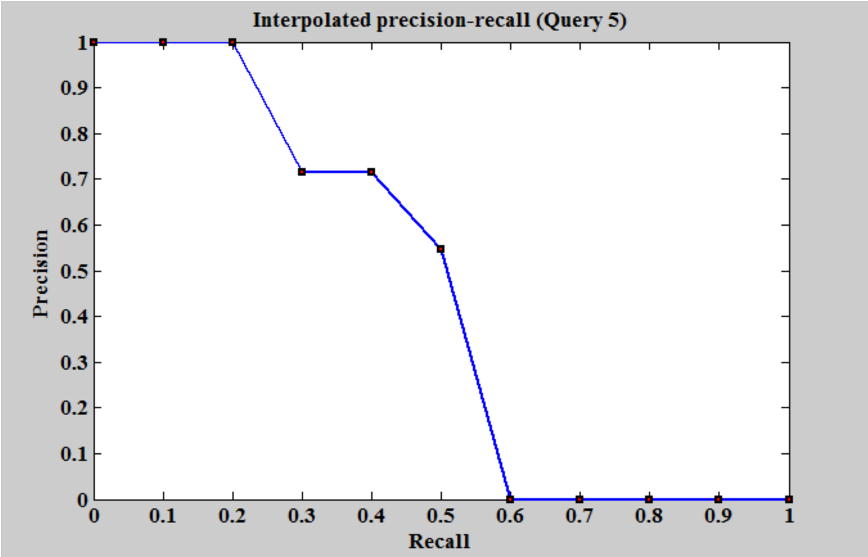

Figure 6.7 The 11-point interpolated precision-recall graph for query 5

❖ ***Query 6:***

In this query, "the client wants to travel to West Europe, like Germany, and is looking for the weather conditions of the respective countries, or geopolitical entities". Of the 29 available web services for this query, 16 web services are discovered that should be ranked by AMW. The graph below also demonstrates that the accuracy of the result is high, but there are two web services for which the input data are unreliable.

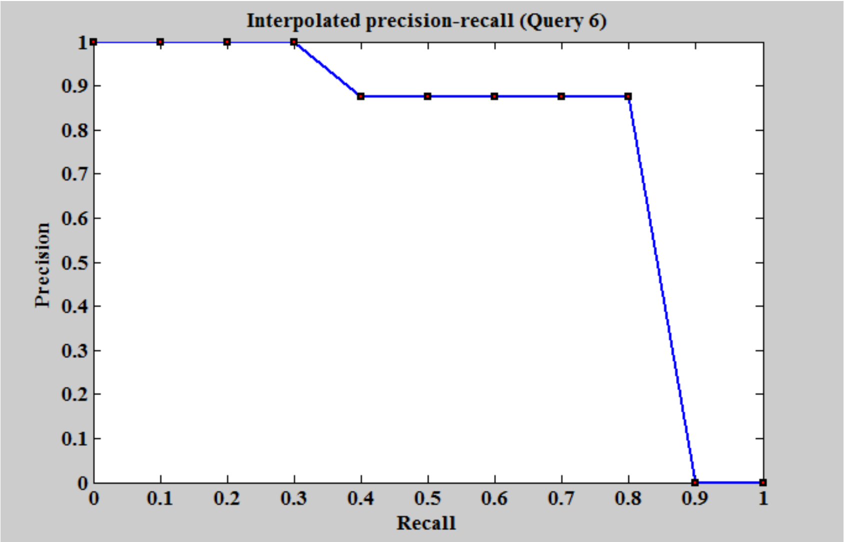

Figure 6.8 The 11-point interpolated precision-recall graph for query 6

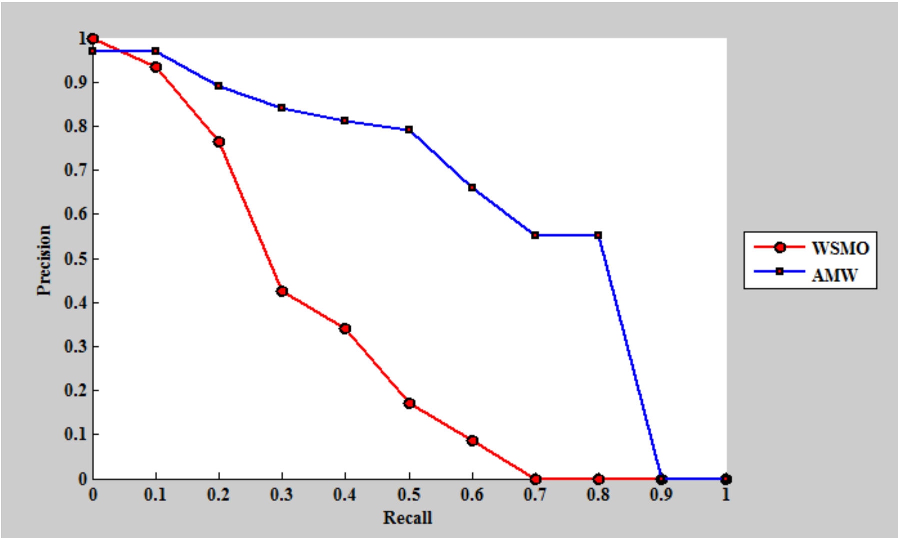

Figure 6.9 Averaged 11-point interpolated precision-recall graph across all queries

As the final experimental result, the averaged 11-point interpolated precision-recall graph across all queries is illustrated. To compare the result with the closest approach, WSMO, the curve for the WSMO result is also added. As mentioned before, when a curve is closer to the top right corner of the graph, its accuracy is higher. It is also considered before that comparisons are best made in three different recall ranges: 0 to 0.2, 0.2 to 0.8, and 0.8 to 1. These ranges characterize high precision, middle recall, and high recall, respectively [84]. The final graph demonstrates that the both approach have good value in the high precision range, but in the middle recall and high recall the results are different. In the middle recall range indeed the situation of AMW is better. The AUC of the curve of AMW is greater than WSMO. However in the high recall range, this difference being decreased.

The other measure of comparison is the area under the curve (AUC). In this graph, the AUC of AMW is greater than that of WSMO. Finally, the MAP values of this graph for AMW and WSMO are 0.6390 and 0.3381, respectively.

6.5 Qualitative Analysis Result

In this section, results provided in Sections 6.1 and 6.2 are analysed. The analysis result is based on the DESMET method described in Chapter 4. Furthermore, the type of analysis is *feature analysis*, as suggested by Barbara Kitchenham [81]. Based on the DESMET method, there are three steps for feature analysis: *feature identification, feature scoring and analysis*. These steps are described below.

6.5.1 Feature Identification

The features for assessment of the approaches are as follows:

- ✓ ***QoS:*** Does the approach use non-functional (QoS) properties for selection?
- ✓ ***Accuracy:*** Does the method select the correct service?

- ✓ ***Extra QoS:*** Are there QoSs in addition to the default QoS described [4]?

- ✓ ***MCDM:*** Does the approach use MCDM? Although the MCDM is a method under the category of decision making in WSS classification, it is also can be a criterion for evaluation of approach.

- ✓ ***CL:*** Does the approach use the CL to assess the rates of alternatives?

- ✓ ***Expert DMs:*** Does the approach use the expert DMs to aggregate the rates of alternatives?

- ✓ ***Default Weights:*** Does the approach propose default criteria weights to the service consumer?

- ✓ ***Automation:*** Does the approach facilitate automatic service selection?

- ✓ ***Flexibility:*** How flexible is the approach with changing criteria and requirements?

6.5.2 Feature Scoring

There are two types of features: *simple* features, which are assessed by a simple YES/NO scale, and *compound* features, for which the degree of support offered by the approach must be measured on an ordinal scale. In this research, *extra QoS, CL, Expert DMs* and *default weights* are simple features, whereas *QoS*, *accuracy*, *MCDM*, *automation* and *flexibility* are compound features.

Each feature should be accompanied by an assessment of the degree of *importance*. Each compound feature must be accompanied by an assessment of its *importance* and *conformance* to a particular feature or characteristic. Scales for measuring importance and conformance are discussed below.

6.5.2.1 Importance

A good approach is one that includes the features that are most important to users. The importance of a feature can be assessed by considering whether it is mandatory or only desirable. This view of importance leads to two assessment criteria: one that identifies whether a feature is mandatory and one that assesses the extent to which a non-mandatory feature is desired. The following scale points must be considered to assess a feature:

- ***M*** Mandatory
- ***HD*** Highly Desirable
- ***D*** Desirable
- ***N*** Nice to have

The importance of features in this research is shown below.

Table 6.54 Importance of features

Feature	*QoS*	*Accuracy*	*Extra QoS*	*MCDM*	*CL*	*Expert DMs*	*Default Weights*	*Automation*	*Flexibility*
Importance	*M*	*HD*	*D*	*HD*	*HD*	*HD*	*D*	*HD*	*D*

Moreover, the importance assessment can be considered as a weighting factor. The following weights are suggested by Barbara Kitchenham:

- Mandatory features: 10
- Highly desirable: 6
- Desirable: 3
- Nice: 1

The importance graph is shown in the figure below.

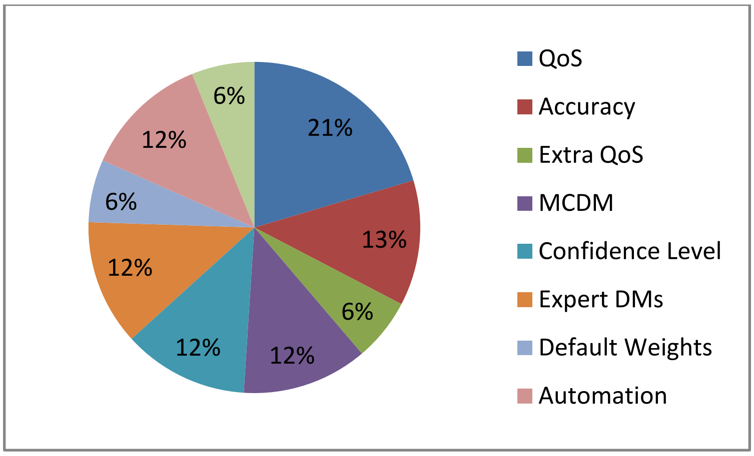

Figure 6.10 Graph of the importance of features

6.5.2.2 Conformance

The aims of the assessment scale for conformance are to define the required level of support of a particular feature and to provide the assessor with a consistent measurement scale against which to score the feature of a particular candidate. The figure below presents the scoring table proposed in [81].

Generic scale point	Definition of Scale point	Scale Point Mapping
Makes things worse	Cause Confusion. The way the feature is implemented makes it difficult to use and/or encouraged incorrect use of the feature	-1
No support	Fails to recognise it. The feature is not supported nor referred to in the user manual	0
Little support	The feature is supported indirectly, for example by the use of other tool features in non-standard combinations.	1
Some support	The feature appears explicitly in the feature list of the tools and user manual. However, some aspects of feature use are not catered for.	2
Strong support	The feature appears explicitly in the feature list of the tools and user manual. All aspects of the feature are covered but use of the feature depends on the expertise of the user	3
Very strong support	The feature appears explicitly in the feature list of the tools and user manual. All aspects of the feature are covered and the tool provides tailored dialogue boxes to assist the user.	4
Full support	The feature appears explicitly in the feature list of the tools and user manual. All aspects of the feature are covered and the tool provides user scenarios to assist the user such as "Wizards".	5

Figure 6.11 Assessment scale used to assess tool support for a feature

The assessment table for AMW and WSMO with respect to the above assessment scale is shown below.

Table 6.55 Assessment table for AMW and WSMO

Feature	*QoS*	*Accuracy*	*Extra QoS*	*MCDM*	*CL*	*Expert DMs*	*Default Weights*	*Automation*	*Flexibility*
AMW	*5*	*4*	*Yes*	*5*	*Yes*	*Yes*	*Yes*	*3*	*4*
WSMO	*5*	*3*	*No*	*3*	*No*	*No*	*No*	*2*	*3*

6.5.3 Analysis

After providing the importance and conformance of features, the score sheets must be analysed and the best approaches determined. Based on the DESMET method, if the acceptance threshold is explicated, the analysis should be based on the difference between the acceptance threshold for each feature set by the users and the score that each approach obtained for the feature. If the acceptance threshold cannot be achieved, the assessment should be based on the scores of the approaches relative to one another. As the acceptance threshold is not achievable in this research, the latter approach is used. Therefore, the analysis must be based on the accumulation of the absolute scores, as described in Section 4.2.4.2.

If there are simple features, a score of “YES” or “NO” must be assigned. Based on the DESMET suggestion, the provision of a simple feature should score five, whereas failure to provide a simple feature should score zero. Moreover, the importance assessment can be used as a weighting factor.

Based on the above descriptions, to express the analysis result, DESMET proposes providing an evaluation profile for each approach both *numerically* and *graphically*. This research uses both approaches. Finally, the overall results of both approaches are discussed.

6.5.3.1 Numerical Evaluation

The presentations of the results based on DESMET via the average evaluation profile for AMW and WSMO are shown below.

Table 6.56 Average evaluation profile for AMW and WSMO

Feature	*Importance*	*AMW*	*WSMO*
QoS	*M*	*5*	*5*
Accuracy	*HD*	*3*	*2*
Extra QoS	*D*	*Yes*	*No*
MCDM	*HD*	*5*	*3*
CL	*HD*	*Yes*	*No*
Expert DMs	*HD*	*Yes*	*No*
Default Weights	*D*	*Yes*	*No*
Automation	*HD*	*3*	*2*
Flexibility	*D*	*4*	*3*

This table is filled based on the description of *importance* and *conformance* discussed in Sections 6.5.2.1 and 6.5.2.2, respectively. The above result is shown with absolute scores, which allows the overall result to also be presented. First, the importance is converted to an absolute score and then aggregated with respect to conformance. For example, the combined score of the flexibility feature for AMW in the above tables is 40, but it would be 120 with respect to importance. The final evaluation profile is shown below.

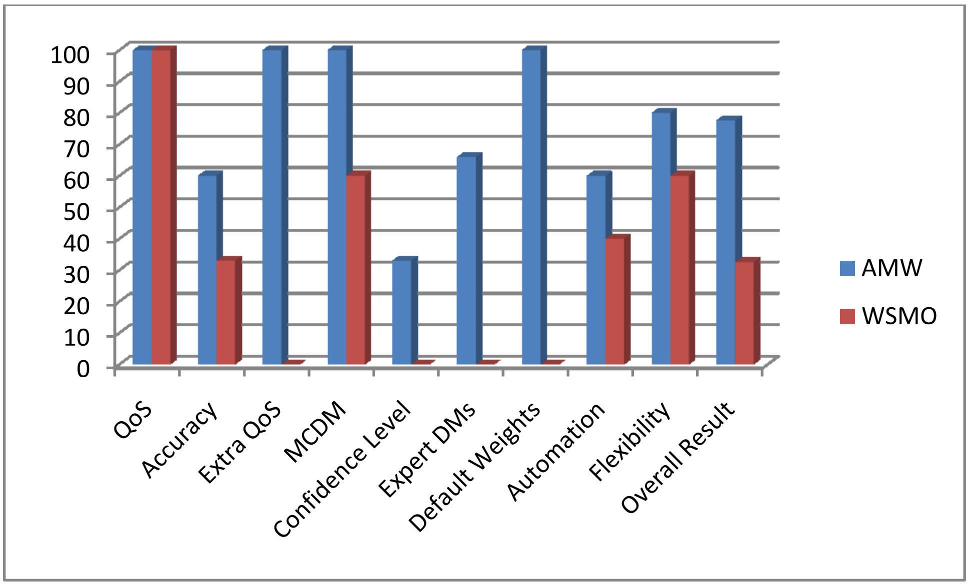

Figure 6.12 Final Evaluation profile

6.5.3.2 Graphical Evaluation

The other way to represent results is to use multiple-metric graphs [85], which are a variation of Kiviat diagrams. The advantage of using multiple-metric graphs is that the data do not need to be converted to absolute scores to compare approaches. All features and approaches are drawn in a graph, which facilitates comparison.

As the final and overall result, the multiple-metric graph for AMW and WSMO based on the data provided in the prior section are shown below.

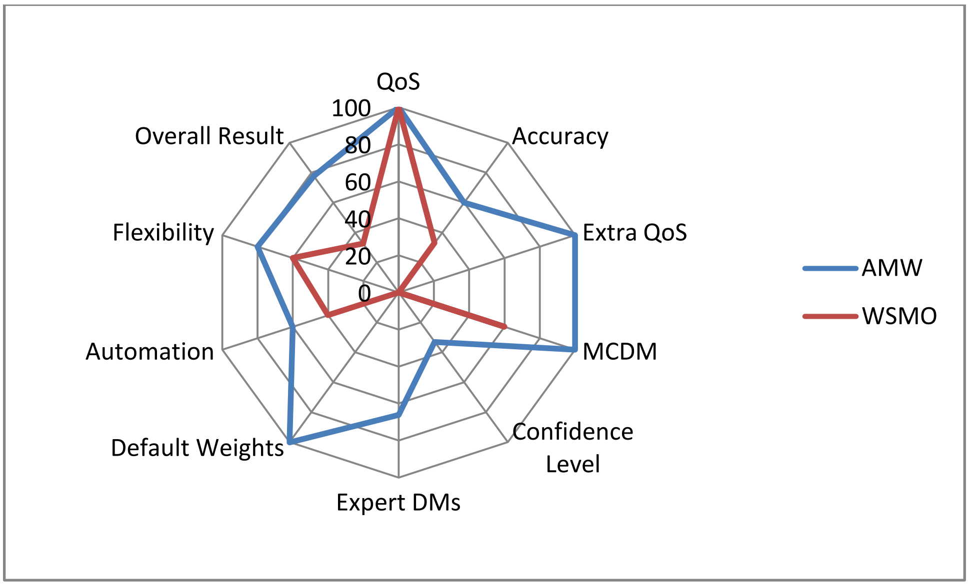

Figure 6.13 Final multiple-metric graph for AMW and WSMO

In the above graphs, which consider all of the states of AMW and WSMO, the improvement of most of the features, such as accuracy, MCDM, default weights and flexibility, is rational. Because AMW employs a proven MCDM method (VIKOR), the CL of the SP, the power of expert DMs and the proposal of default criteria weights improve each feature.

6.6 Summary

In this chapter, the validation and evaluation of the AMW approach are presented in terms of quantitative and qualitative. The results of the case studies and experiments are quantitatively evaluated using the accuracy measurement, precision-recall, which was adopted for ranked retrievals. The results are separated into the results of case studies and the experimental results of AMW on the created test collection. For convenience and a better understanding of the results, the MAP is considered. Moreover, the final result of AMW is compared with the result of WSMO, which is the previous method most similar to the proposed approach. This comparison shows that AMW has greater accuracy than WSMO.

The qualitative manner of AMW presented using the DESMET method. The proposed approach is validated with two case studies. The results demonstrate the proposed features by AMW. The types of problems that can be solved by AMW but not by the existing approaches are investigated. The approach is validated analytically. The case studies in which some scenarios are defined facilitate the validation of AMW. The evaluation of the approach is performed using the *feature analysis* proposed by Barbara Kitchenham. The result of the investigative evaluation demonstrates that AMW satisfies every feature, i.e., QoS, MCDM, flexibility, accuracy, automation, CL and default weight. The results of the feature analysis evaluation are expressed numerically and graphically. The improvements of those features are sensible compared with the existing approaches. Furthermore, the results of this chapter represent the achievement of the objectives of this research.

CHAPTER 7

CONCLUSION

This chapter presents the conclusions of this research. The achievements and contributions of the proposed approach, AMW, are summarised, and unresolved issues and future work are described.

7.1 Summary and Achievement

In this section, the research is summarised and achievements are discussed. The objectives, which are described in Chapter 1, are investigated individually. Additionally, this section discusses how this research achieves the objectives.

Objective (i): To investigate and evaluate the state of the art in WSS approaches.

To achieve to this objective, first, the existing approaches in WSS are investigated. Second, those approaches are compared with respect to the defined criteria for WSS, which are described in Chapter 3. The comparison is performed on two levels; because the research is based on WSMO and MCDM, the second level is also broken down into the semantic level and MCDM level. In the first level, the general approaches of service selection based on modelling, collection data and decision making are investigated. In the second level, the depth of service selection modelling and decision making are investigated. The aim of the investigation and evaluation of the existing approaches is to clarify the weaknesses and strengths of the current approach.

Objective (ii): To develop a new approach for WSS in which the accuracy of both the input data and selection method are considered.

Once objective (i) is achieved and the weaknesses of current approaches are investigated, the main gap of the research is the lack of consideration of the accuracy of both the input data and selection method. Therefore, to bridge this research gap, an approach is proposed in which the accuracy of the selection method is considered and improved while utilising the accuracy of the input data, which is expressed by the service provided. As SPs advertise their own services, realism and accuracy should be considered.

Objective (iii): To propose default criteria weights using the AHP method to help users express their preferences.

One of the essential components of service selection is the weighting of criteria. Typically, the criteria weights are gathered from the service consumer. This is not always the case because the user is often unfamiliar with the weighting system and do not know how to establish the criteria weights. Therefore, the default criteria weights are calculated based on the opinions of experts in the area of web services. Then, the default weights are suggested to users. If a user is familiar with the weighting system and wants to modify the criteria weights, the user can do so. In this situation, the AMW avoids any incorrect weighting of criteria due to the user's lack of knowledge in this area.

Objective (iv): To design and formulate algorithms to support the proposed approach.

Six algorithms are proposed to support the proposed approach. These algorithms support the dynamic aspect and particularly the service selection mechanism described in Section 5.3.3. As described in that section, there are three stages in AMW for service selection, namely, *pre-selection, selection* and *post-selection*. To support the pre-selection stage, two algorithms, which involve defining the default criteria weights and providing rates of alternatives, are proposed. Algorithms 1 and 2 are related to this stage. The basis of AMW is the selection algorithm, in which the decision making is performed. Algorithm 3 is used in this stage. The post-selection stage includes two algorithms, Algorithms 4 and 5. One algorithm estimates the CL of the SP, and the other estimates the reputation. Finally, to improve the automation of the AMW approach, the goal file is generated automatically based on user preferences. Algorithm 6 is proposed to generate the goal file.

Objective (v): To evaluate the accuracy of the proposed approach by developing a selector tool and comparing it with other approaches.

To evaluate this research, AMW is used for two case studies to validate the applicability of the proposed approach on WSS. Furthermore, the application of the case study specifies the new features of the AMW that the existing approaches do not consider. Then, the AMW approach is evaluated based on the result of the experiment performed on the AMW_TC1, which involves 197 services and six queries. The experimental result is used to improve the accuracy measurement. The accuracy measurement is based on the precision and recall concept discussed in Chapters 4 and 5. Finally, the result of the case studies and experiments are analysed based on feature analysis and accuracy measurement. The former is qualitative validation, and the latter is quantitative validation.

7.2 Contributions of the Research

The contributions of this research can be summarised as follows:

(i) Default criteria weights for WSS are obtained in this research using the AHP method.

(ii) To improve the accuracy of result and input data, two features are added via AMW. To avoid any inaccurate data from the SP, the CL of the SP and the power of the DMs are considered.

(iii) The enhanced VIKOR is proposed to eliminate the shortcomings of the VIKOR method.

(iv) To compensate for the shortcomings of the existing approaches, the AMW architecture is proposed and six supportive algorithms for the proposed approach are formulated.

(v) The automatic goal generation method and its supportive algorithm, which converts the expressed user preferences to the format of the goal WSML file, are proposed.

(vi) To support the architectural and algorithmic aspects of proposed approach, the AMW tool is implemented in the Java programming language.

7.3 Future Work

The future work and open issues of this research are as follows:

(i) The discovery stage is one stage before selection. The results of discovery can affect the results of the selection stage. For example, suppose that after discovery and selection, the service C must be discovered and selected from the list of services A, B, C and D. If the discovery stage cannot successfully discover service C, the result of the selection stage is affected even though the selection method is accurate. Therefore, integrating the discovery stage and selection stage is an unresolved issue of this research that can be considered in future work.

(ii) The goal generator proposed in this research converts and adds user preferences to the non-functional properties section in the WSML file and is sufficient for this research. However, there are many services available in the other format, and there are not any tools to convert these services to the WSML format. The creation of such a convertor can be considered in future work.

(iii) The integration of the AMW prototype with WSMX as the execution environment of WSMO will also be investigated because this integration can improve the selection component of WSMX as a commercial tool for WSS. Furthermore, because AMW and WSMX are based on Java, the integration is straightforward. To achieve the research results and integrate them as a commercial tool, AMW can be integrated with WSMX.

REFERENCES

1. Haas, H. and A. Brown. Web Services Glossary. 2004 11 February 2004]; Available from: http://www.w3.org/TR/ws-gloss/.
2. Tabatabaei, S.G.H., W.M.N.W. Kadir, and S. Ibrahim. A comparative evaluation of state-of-the-art approaches for web service composition. in Software Engineering Advances, 2008. ICSEA'08. 2008: IEEE.
3. de Oliveira Jr, F.G.A. and J.M.P. de Oliveira, QoS-based Approach for Dynamic Web Service Composition. Journal of Universal Computer Science, 2011. 17(5): p. 712-741.
4. Fensel, D., M. Kerrigan, and M. Zaremba, Implementing Semantic Web Services. 2008, Berlin: Springer.
5. Sheng, Q.Z., et al., Configurable composition and adaptive provisioning of web services. Services Computing, IEEE Transactions on, 2009. 2(1): p. 34-49.
6. Zhao, P. and L. Di, Geospatial Web Services: advances in information interoperability. 2011: Information Science Publishing.
7. da Silva, E.G., L.F. Pires, and M. van Sinderen, Towards runtime discovery, selection and composition of semantic services. Computer Communications, 2011. 34(2): p. 159-168.
8. Cheng, D.Y., et al., A user centric service-oriented modeling approach. World Wide Web, 2011. 14(4): p. 431-459.
9. Liu, Y., A.H.H. Ngu, and L. Zeng. QoS computation and policing in dynamic web service selection. 2004. New York, NY, United states: Association for Computing Machinery.
10. Pan, Z. and J. Baik, A QOS enhanced framework and trust model for effective web services selection. Journal of Web Engineering, 2010. 9(4): p. 327-346.
11. Kerrigan, M. Web service selection mechanisms in the web service execution environment (WSMX). in Proceedings of the 2006 ACM symposium on Applied computing. 2006: ACM.
12. Klusch, M., B. Fries, and K. Sycara, OWLS-MX: A hybrid Semantic Web service matchmaker for OWL-S services. Web Semantics: Science, Services and Agents on the World Wide Web, 2009. 7(2): p. 121-133.

13. Klusch, M., P. Kapahnke, and I. Zinnikus. Hybrid adaptive web service selection with SAWSDL-MX and WSDL-analyzer. 2009. Heraklion, Crete, Greece: Springer Verlag.
14. Li, L., Y. Wang, and E.P. Lim, Trust-Oriented Composite Service Selection with QoS Constraints. Journal of Universal Computer Science, 2010. 16(13): p. 1720-1744.
15. OWL-S: Semantic Markup for Web Services. 22 November 2004; Available from: http://www.w3.org/Submission/OWL-S.
16. Kopecky, J., et al., Sawsdl: Semantic annotations for wsdl and xml schema. Internet Computing, IEEE, 2007. 11(6): p. 60-67.
17. Booth, D., et al. Web Services Architecture. 11 February 2004; Available from: http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/.
18. New to SOA and Web services. Available from: http://www.ibm.com/developerworks/webservices/newto/websvc.html.
19. Fensel, D., et al., Enabling Semantic Web Services. 2007, Berlin: Springer.
20. Christensen, E., et al. WSDL v1.1. 15 March 2001; Available from: http://www.w3.org/TR/wsdl.
21. SOAP v1.2. 2007; Available from: http://www.w3.org/TR/2007/REC-soap12-part1-20070427/
22. Web Services Description Language (WSDL) Version 2.0 2003; Available from: http://www.w3.org/TR/2003/WD-wsdl20-20031110/.
23. Mitra, S., I/O-Automata based formal approach to Web Services Choreography. 2009, Iowa State University.
24. Berners-Lee, T., J. Hendler, and O. Lassila, The Semantic Web. Scientific American, 2001: p. 34–43.
25. Timm, J.T.E., A model-driven framework for the specification, grounding, and execution of semantic Web services. April 2008, Arizona State University.
26. Smith, M.K., C. Welty, and D.L. McGuinness. Owl web ontology language guide. February 2004; Available from: http://www.w3.org/TR/owl-guide.
27. Triantaphyllou, E., et al., Multi-criteria decision making: an operations research approach. Encyclopedia of electrical and electronics engineering, 1998. 15: p. 175-186.
28. Saaty, T.L., Analytic hierarchy process. 1980: Wiley Online Library.

29. Russell, R.S.a.T.I., Bernard W. , ed. Operations Management 2003: New Jersey.

30. Coyle, G., The analytic hierarchy process (AHP), in Practical Strategy: Structured Tools and Techniques, Open Access Material. 2004, Pearson Education Ltd Glasgow.

31. Kunz, J., The Analytic Hierarchy Process (AHP). Eagle City Hall Location Options Task Force, 2010.

32. Yoon, K.P. and C.L. Hwang, Multiple attribute decision making: an introduction. 1995: Sage Publications, Incorporated.

33. Hwang, C.-L. and K. Yoon, Multiple attribute decision making : methods and applications ; a state-of-the-art survey. 1981, Berlin [u.a.]: Springer.

34. Opricovic, S. and G.-H. Tzeng, Compromise solution by MCDM methods: A comparative analysis of VIKOR and TOPSIS. European Journal of Operational Research, 2004. 156(Compendex): p. 445-455.

35. Opricovic, S., Multicriteria optimization of civil engineering systems. Faculty of Civil Engineering, Belgrade, 1998. 2(1): p. 5-21.

36. Rao, J. and X. Su. A survey of automated Web service composition methods. 2005. San Diego, CA, United states: Springer Verlag.

37. Manish, G., S. Rajendra, and M. Shrikant, Web Service Selection Based on Analytical Network Process Approach, in Proceedings of the 2008 IEEE Asia-Pacific Services Computing Conference. 2008, IEEE Computer Society.

38. Zou, G., et al. An agent-based web service selection and ranking framework with QoS. 2009. Beijing, China: IEEE Computer Society.

39. Mani, A. and A. Nagarajan. Understanding quality of service for Web services. Jan, 2002; Available from: http://www-106.ibm.com/developenvorksllibrary/ws-quality.html.

40. Menasc, D.A., QoS Issues in Web services. IEEE Internet Computing, Nov./Dec. 2002. 6: p. 72-75.

41. Yu, T., Quality of Service (QoS) in Web services: Model, Architecture and Algorithms, in Electrical and Computer Engineering. 2006, University of California.

42. Huang, A.F.M., C.-W. Lan, and S.J.H. Yang, An optimal QoS-based Web service selection scheme. Information Sciences, 2009. 179(19): p. 3309-3322.

43. Menasce, D.A., QoS issues in web services. IEEE Internet Computing, 2002. 6(6): p. 72-75.

44. Yu, T. and K.-J. Lin, Service selection algorithms for Web services with end-to-end QoS constraints. Springer Journal of Information Systems and E-Business Management, June 10, 2005: p. 103-126.

45. O'Sullivan, J., D. Edmond, and A. Ter Hofstede, What's in a Service? Distributed and Parallel Databases, 2002. 12(2): p. 117-133.

46. Janicke, H. and M. Solanki. Policy-driven service discovery. in 2nd European Young Researchers Workshop on Service Oriented Computing. 2007.

47. Yu, H.Q. and S. Reiff-Marganiec, Non-functional Property based service selection: A survey and classification of approaches. 2008, Sun SITE Central Europe.

48. Wang, Y. and J. Vassileva. A review on trust and reputation for web service selection. 2007. Toronto, ON, Canada: Institute of Electrical and Electronics Engineers Inc.

49. Galizia, S., A. Gugliotta, and J. Domingue. A trust based methodology for web service selection. 2007. Irvine CA, United states: Inst. of Elec. and Elec. Eng. Computer Society.

50. Page, L., et al., The pagerank citation ranking:Bringing order to the web. 1998, Stanford Digital Library Technologies Project.

51. Domingue, J., et al., IRS-III: A broker-based approach to semantic Web services. Web Semantics, 2008. 6(2): p. 109-132.

52. Zuo, M., S. Wang, and B. Wu. Research on web services selection model based on AHP. in Service Operations and Logistics, and Informatics, IEEE/SOLI 2008. 2008. Beijing, China: IEEE.

53. Seo, Y.-J., H.-Y. Jeong, and Y.-J. Song. A study on web services selection method based on the negotiation through quality broker: A MAUT-based approach. 2005. Hangzhou, China: Springer Verlag.

54. Al-Masri, E. and Q.H. Mahmoud. Discovering the best web service: A neural network-based solution. 2009. San Antonio, TX, United states: Institute of Electrical and Electronics Engineers Inc.

55. Feier, C., et al. Towards intelligent web services: Web service modeling ontology (WSMO). in the International Conference on Intelligent Computing (ICIC 2005).

56. WSMO: Web Service Modeling Ontology. 3 June 2005; Available from: http://www.w3.org/Submission/WSMO.
57. WSML Language Reference. 2008-08-08; Available from: http://www.wsmo.org/TR/d16/d16.1/v1.0/.
58. Web Service Modeling Language (WSML). 3 June 2005; Available from: http://www.w3.org/Submission/WSML/.
59. Gekas, J. Web Service Ranking in Service Networks. in The 3rd European Semantic Web Conference. June 11-14, 2006. Budva, Montenegro.
60. Paolucci, M., et al., Semantic Matching of Web Service Capabilities, in Horrocks, I. & J. Hendler (eds.): 1st International Semantic Web Conference. 2002.
61. Maheswari, S. and G. Karpagam, QoS Based Efficient Web Service Selection. European Journal of Scientific Research, 2011. 66(3): p. 428-440.
62. Manikrao, U.S. and T.V. Prabhakar. Dynamic selection of web services with recommendation system. 2005. Seoul, Korea, Republic of: Inst. of Elec. and Elec. Eng. Computer Society.
63. Doshi, P., R. Goodwin, and R. Akkiraju, Parameterized Semantic Matching for Workflow Composition. March 2004, IBM.
64. Klusch, M. and P. Kapahne. Semantic Web Service Selection with SAWSDL-MX. in Second International Workshop on Service Matchmaking and Resource Retrieval in the Semantic Web. 2008. Germany.
65. Wang, P., K.M. Chao, and C.C. Lo, On optimal decision for QoS-aware composite service selection. Expert Systems with Applications, 2010. 37(1): p. 440-449.
66. Wang, P., et al. A fuzzy model for selection of QoS-aware web services. in IEEE International Conference on e-Business Engineering, ICEBE 2006. 2006. Shanghai, China: Inst. of Elec. and Elec. Eng. Computer Society.
67. Qu, L.-l. and Y. Chen. QoS ontology based efficient web services selection. in Management Science and Engineering, ICMSE 2009. 2009.
68. Lo, C.-C., et al. Service selection based on fuzzy TOPSIS method. in 24th IEEE International Conference on Advanced Information Networking and Applications Workshops, WAINA 2010. 2010. Perth, Australia: IEEE Computer Society.

69. Godse, M., R. Sonar, and S. Mulik. Web Service Selection Based on Analytical Network Process Approach. in Asia-Pacific Services Computing Conference, APSCC'08. 2008: IEEE.

70. Karim, R., D. Chen, and C. Chi-Hung. An Enhanced PROMETHEE Model for QoS-Based Web Service Selection. in Services Computing (SCC), 2011 IEEE. 2011.

71. Toma, I., et al., A Multi-criteria Service Ranking Approach Based on Non-Functional Properties Rules Evaluation, in Service-Oriented Computing–ICSOC 2007, B. Krämer, K.-J. Lin, and P. Narasimhan, Editors. 2007, Springer Berlin Heidelberg. p. 435-441.

72. Garcıa, J.M., et al. A service ranker based on logic rules evaluation and constraint programming. 2008: Citeseer.

73. Grossman, D.A. and O. Frieder, Information retrieval: Algorithms and heuristics. Vol. 15. 2004: Springer.

74. Yu, H.Q. and S. Reiff-Marganiec. A method for automated web service selection. 2008: IEEE.

75. Wang, X., et al., A qos-aware selection model for semantic web services. Service-Oriented Computing–ICSOC 2006, 2006: p. 390-401.

76. Lazaro, M. and E. Marcos. Research in Software Engineering: Paradigms and Methods. in 17th International Conference on Advanced Information System (CAiSE'05). June 2005. Porto, Portugal.

77. Adrion, W.R. Research methodology in software engineering. in Summary of the Dagstuhl Workshop on Future Directions in Software Engineering" Ed. Tichy, Habermann, and Prechelt, ACM Software Engineering Notes, SIGSoft. 1993.

78. Zelkowitz, M.V. and D.R. Wallace, Experimental models for validating technology. Computer, 1998. 31(5): p. 23-31.

79. Zelkowitz, M.V., D.R. Wallace, and D.W. Binkley, Experimental validation of new software technology. Series on Software Engineering and Knowledge Engineering, 2003. 12: p. 229-263.

80. Rijsbergen, C.J.V., Information Retrieval. 1979: Butterworth-Heinemann. 208.

81. Kitchenham, B., S. Linkman, and D. Law, DESMET: a methodology for evaluating software engineering methods and tools. Computing & Control Engineering Journal, 1997. 8(3): p. 120-126.

82. Kitchenham, B.A., Evaluating software engineering methods and tool\—part 3: selecting an appropriate evaluation method\—practical issues. SIGSOFT Softw. Eng. Notes, 1996. 21(4): p. 9-12.
83. Manning, C.D., P. Raghavan, and H. Schütze, Introduction to information retrieval. Vol. 1. 2008: Cambridge University Press Cambridge.
84. Harman, D.K., 4th Text Retrieval Conference. 1998: Diane Publishing Company.
85. Pfleeger, S.L., J.C. Fitzgerald, and D.A. Rippy, Using multiple metrics for analysis of improvement. Software Quality Journal, 1992. 1(1): p. 27-36.
86. Zaremba, M., et al., Wsmx architecture. WSMX Deliverable D, 2005. 13.
87. Herold, M., WSMX documentation. Digital Enterprise Research Institute Galway, Ireland, 2008. 3.
88. Haller, A., et al. Wsmx-a semantic service-oriented architecture. 2005: IEEE.
89. Chen, S.J., C.L. Hwang, and F.P. Hwang, Fuzzy multiple attribute decision making:(methods and applications). 1992: Springer.

APPENDIX A

LIST OF PUBLICATIONS

Journal Papers:

- Mojtaba Khezrian, Wan M. N. Wan Kadir, Suhaimi Ibrahim, Alaeddin Kalantari: Service Selection based on VIKOR method. International Journal of Research and Reviews in Computer Science Volume 2, Issue 5, UK, 2011.

- Mojtaba Khezrian, Wan M. N. Wan Kadir, Suhaimi Ibrahim, Alaeddin Kalantari: A Hybrid Approach for Web Service Selection. International Journal of Computational Engineering Research (IJCER) Volume 2, Issue 1, 2012.

- Mojtaba Khezrian, Wan M. N. Wan Kadir, Suhaimi Ibrahim: An Efficient Approach for Service Selection Based on MCDM and QoS. International Journal On Advances in Information Sciences and Service Sciences, Volume 5, Issue 7 , S.Korea, 2013. (SCOPUS)

Conference Papers:

- Mojtaba Khezrian, Wan M. N. Wan Kadir, Suhaimi Ibrahim, Keyvan Mohebbi, Kanmani Munusamy, Sayed Gholam Hassan Tabatabaei iiWAS, An evaluation of state-of-the-art approaches for web service selection '10 Proceedings of the 12th International Conference on Information Integration and Web-based Applications & Services, 2010.

- Mojtaba Khezrian, Wan M. N. Wan Kadir, Suhaimi Ibrahim, Evaluation of Current Web Service Selection Approaches, The 6th FSKSM Postgraduate Annual Research Seminar (PARS '10), 2010, Skudai, Malaysia.