Automatic Message Annotation and Semantic Inference for Context Aware Mobile Computing

A Thesis submitted for the Degree of
Doctor of Philosophy

By
Ghaidaa A. B. Al-sultany

Electronic and Computer Engineering
School of Engineering and Design
Brunel University
UK

May 2012

AUTHOR'S DECLARATION

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners. I understand that my thesis may be made electronically available to the public; therefore I authorise Brunel University to make available electronically to individual or institutions for the purpose of scholarly research.

Signature: ....

Ghaidaa Al-Sultany

ABSTRACT

In this thesis, the concept of mobile messaging awareness has been investigated by designing and implementing a framework which is able to annotate the short text messages with context ontology for semantic reasoning inference and classification purposes. The annotated metadata of text message keywords are identified and annotated with concepts, entities and knowledge that drawn from ontology without the need of learning process and the proposed framework supports semantic reasoning based messages awareness for categorization purposes.

The first stage of the research is developing the framework of facilitating mobile communication with short text annotated messages (SAMS), which facilitates annotating short text message with part of speech tags augmented with an internal and external metadata. In the SAMS framework the annotation process is carried out automatically at the time of composing a message. The obtained metadata is collected from the device's file system and the message header information which is then accumulated with the message's tagged keywords to form an XML file, simultaneously. The significance of annotation process is to assist the proposed framework during the search and retrieval processes to identify the tagged keywords and The Semantic Web Technologies are utilised to improve the reasoning mechanism.

Later, the proposed framework is further improved "Contextual Ontology based Short Text Messages reasoning (SOIM)". SOIM further enhances the search capabilities of SAMS by adopting short text message annotation and semantic reasoning capabilities with domain ontology as Domain ontology is modeled into set of ontological knowledge modules that capture features of contextual entities and features of particular event or situation. Fundamentally, the framework SOIM relies on the hierarchical semantic distance to compute an approximated match degree of new set of relevant keywords to their corresponding abstract class in the domain ontology. Adopting contextual ontology leverages the framework performance to enhance the text comprehension and message categorization.

Fuzzy Sets and Rough Sets theory have been integrated with SOIM to improve the inference capabilities and system efficiency. Since SOIM is based on the degree of similarity to choose the matched pattern to the message, the issue of choosing the best-retrieved pattern has arisen during the stage of decision-making. Fuzzy reasoning classifier based rules that adopt the Fuzzy Set theory for decision making have been applied on top of SOIM framework in order to increase the accuracy of the classification process with clearer decision. The issue of uncertainty in the system has been addressed by utilising the Rough Sets theory, in which the irrelevant and indecisive properties which affect the framework efficiency negatively have been ignored during the matching process.

ACKNOWLEDGMENT

"***Whoever relies upon Allah, He is sufficient for him***"

I am grateful to the creator who's always close to me in all situations. God, my thankfulness and greatest glory to support me conducting this research work to completion.

I would like to express my sincere appreciation and gratitude to all those who made this thesis possible. This work wouldn't have been possible without help, support and inspiration of my supervisors, Dr. Maozhen Li and Professor Hamid Al-Raweshidy. I am heartily thankful to Dr. Li, whose encouragement, guidance and support from the beginning till end made my work much easier. I appreciate all his help and support.

Big thanks go to my colleagues in the CMCR Centre, starting with the head of centre Professor Abdul H. Sadka for embracing us with his advices and kindness. Thank you Amal, Rafiq, Sadiq, Obaid, Nawaz, Sanusi, Mohib, Abdulqadir and Ibrahim for the unforgettable warm atmosphere. You are indeed valued friends.

I am indebted to show my gratitude to Dr. Sadaqat Jan , Dr. Suhel Hammoud and Dr. Ibrar Shah, who supported me during the completion of the project.

My appreciation to the Ministry of Higher Education and Scientific Research (IRAQ) for sponsoring this research work.

My deepest gratitude to my sisters and brothers for their tremendous encouragement and prayers that always guided me to the right way.

Last of all, my sincere appreciation to all of those who supported me attaining my project successfully.

DEDICATION

I dedicate this thesis to the dearest people in my life...

- *My parents – for deep and unending love.*
- *My husband – for sacrifices, patience and motivation*
- *Mohammed and Abdullah – To be proud of your mum.*

TABLE OF CONTENTS

List of Figures

List of Tables

List of Abbreviations

NLP	Natural Language Processing
IR	Information Retrieval
XML	Extensible Markup Language
DTD	Document type Definition
CLDC	connected limited device configuration
JSR	Java specification requests
JVM	Java virtual machine
KVM	Kilo byte virtual machine
CDC	Connected device configuration
DAML	Darpa Agent Markup Language
FL	Fuzzy Logic
SAMS	Facilitating mobile communication with annotated messages
JME	Java Micro Edition
JCP	Java Community Process
W3C	World Wide Web Consortium
FLR	Fuzzy Lattice Reasoning
FST	Fuzzy Set Theory
kXML	Kilo Byte Extensible Markup Language
MIDP	Mobile Information Device Profile
API	Application Programming Interface

List of Figures

OWL	Web Ontology Language
SOIM	Contextual Ontology based Short Text Messages reasoning
RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
URI	Uniform Resource Identifier
PIM	personal Information Management

List of Publications

Journal Papers:

- G. Al-Sultany, M. Li, S. Jan, H. S. Al-Raweshidy, 'Enhancement semantic messages inferences using Fuzzy and Rough Sets Theory,' IEEE Transactions on Mobile Computing (under preparation).
- G. Al-Sultany, M. Li, S. Jan, H. S. Al-Raweshidy, 'Semantic Ontology based Intelligent Mobile Messaging', Computing and Informatics (under review).
- S. Jan, M. Li, G. Al-Sultany, H. S. Al-Raweshidy, I. Shah, 'Semantic file annotation and retrieval on mobile devices,' Mobile Information Systems 7(2), pp. 107-122 (2011).

Conferences Papers:

- G. Al-Sultany, M. Li, S. Jan, H. Al-Raweshidy, 'Mobile Message aware Enhancement using Fuzzy Lattice Reasoning,' IEEE FSKD 2012 (Accepted).
- G. Al-Sultany, M. Li, S. Jan, H. Al-Raweshidy, 'Intelligent mobile messaging. IEEE FSKD 2011: 2671-2674.
- S. Jan, M. Li, G. Al-Sultany, H. S. Al-Raweshidy, 'Semantic based file retrieval on resource limited devices with ontology alignment support,' IEEE FSKD 2011: 2699- 2702.
- S. Jan, M. Li, G. Al-Sultany, H. S. Al-Raweshidy, 'Ontology based file retrieval framework for resource limited mobile devices,' IEEE FSKD 2011: 2687-2690.
- S. Jan, M. Li, G. Al-Sultany, H. S. Al-Raweshidy, 'Ontology alignment using Rough Sets,' IEEE FSKD 2011: 2683-2686.
- G. Al-Sultany, M. Li, S. Jan, H. S. Al-Raweshidy, 'Facilitating Mobile Communication with Annotated Messages,' IEEE CIT 2010: 755-760.
- S. Jan, M. Li, G. Al-Sultany, H. S. Al-Raweshidy, File annotation and sharing on Low- end mobile devices,' IEEE FSKD 2010: 2973-2977.

CHAPTER 1

1 INTRODUCTION

1.1 Overview

The rapid expansion in the information resources has increased the need to retrieve, classify and infer data more efficiently. It has been proven to be an intensive research challenge for the research community in recent years due to the multiple constraints for reaching the efficiency objective. Annotation with additional information and descriptions is one of the most popular methods that have been adopted in the textual information management. Metadata is the key way to keep the resources easily accessible by describing, explaining and managing the information resources [37] [36] . It is used to describe documents, provide access to the information resources, enhance searching, information retrieval and understanding[36] [38] . Knowledge exploitation about information resources has encouraged many semantic awareness applications such as Semantic Web to adopt Metadata in their semantic annotation processes [86] [88]

The current two trends in the field of semantic awareness with respect to semantic annotation are: manual or semiautomatic and automatic annotation mechanism via using different approaches such as ontology description languages for semantic documents formation [82] [83] [85] . Text annotation with ontology concepts is considered as one of the fundamental approaches in developing intelligent Web applications[10] . Basically, it adopts a semi-automated method by initiating a set of manual annotations and then suggesting new additional annotations automatically to the user for extending the metadata in the text [87] [10] .

The rapid evolution and growth of communication technologies revealed different forms of short pieces of texts such as mobile text messages, instant messages and twitter messages [7] . Thus, recent research has focused on short text processing applications such as analysis [10] , classification [4] and annotation [8] [5] . However, the sparse nature and limited information have become a challenging task in short text processing. Most traditional techniques like Bag Of Words (BOW) [111] , standard Natural Language Processing (NLP) and traditional similarity measures, which mainly focus on the word frequency and word co-occurrences, usually fail to achieve desired accuracy due to the absence of sufficient knowledge about the text itself [9] [6] [110] Therefore, integrating short text with meta-information from external knowledge sources like WordNet and Wikipedia are adopted in most existing work. However, online querying are required by these approaches and are extremely time consuming which make them unfit for real time applications [112] .

The utilization of knowledge to find associated entities and concepts of a short text using ontology's structural and semantic relationships, in a given domain, improves reasoning and categorization of the process [82] . In addition automated text annotation based on semantic ontology can be one of the approaches to achieve automated reasoning and permit for extra knowledge to be inferred automatically[10] . In this research, we exploit the technology of semantic awareness to facilitate the notion of ontology based mobile short text message awareness on resource-limited devices.

This chapter describes the background to the problem investigated in this thesis, the motivation and aim of the research, major contributions and the research methodology. Finally, the structure of the thesis is outlined.

1.2 Motivation for Short text Message Awareness

Sufficient research efforts have been carried out by the mobile phone researchers towards the objective of the proficient smart mobile environment. Developing semantic reasoning, service discovery and information retrieval to dynamically and precisely adapt to the mobile user`s reactions are reported as a the challenges and research issues in recent literature [1] [17] . Semantic content awareness and context awareness computing is a core part in information management. It has been intensively investigated during the last decades to obtain, utilize and adapt contextual information and to provide users efficient services seamlessly[17] [18] [19] .

Mobile messaging is one of the most popular services in ubiquitous networks environment that have been embraced by users since the first commercial use of mobile textual messaging services [20] [28] . The cost, efficiency and privacy have encouraged the text messages reciprocation, particularly in social communication networks[29] . The number of mobile users using text messaging in the United States of America from 2003-2010 has increased from about 32 million to 100 million costumers, respectively[133] .

Similarly, on average, over 48 billion messages are sent monthly[30] . In the United Kingdom, the outgoing messages were more than 6.5 billion averaged over a period of one month [31] . It is shown that most of messages are concentrating on some occasions and festivities such as New Year, Christmas Eve and Birthdays. During the Chinese New Year in 2007, about 15 billion messages had been recorded which were being exchanged among users [20] [133] .

The widespread usage of short text messaging system – SMS- reflects its importance for the mobile phone users. However, user interaction with every single message can be affected due to the user's situation and an increasing high volume of short text messages received in special events like festival [11] [12] [20] . The study submitted in [20] stated that the generic greeting messages for example 'happy new year' are less likely to be responded immediately and the recipients of these messages usually use a predefined template to reply to the messages [133] . As the mobile services are evolving, additional features have been added to the short text messages SMS such as multimedia facilities. Thus, it is highly unlikely that the characteristics of SMS will remain unchanged and therefore a further potential stage of adding context awareness in the evolution of text based messaging has been investigated in different studies [12] [11] .

Evidently, the user willingness to read a message is changing given to a specific situation [11] [12] [20] . Since most context awareness applications concentrate on a user context, (*e.g,* profile, location, calendar) in a particular situation to enable the device adaptation to the current context and to infer an appropriate decision accordingly. Short messages can be programmed to be users profile specific, for example, if a user is in a business meeting and he is not supposed to be disturbed receiving messages except the one related with his work. In this case, depending on only the user location, profile or calendar may not help the system to trigger the right action, when a new business message is received. Therefore, the issue of short text messages classification and semantic reasoning may leverage the devices' context awareness and enhance a decision making accuracy.

The limited, sparse and unstructured nature of short text messages have resulted in a difficulty to mine their significant statistics which are otherwise available when texts are long [5] . Therefore, they have been tackled in most current approaches on short text classification and reasoning via employing an external knowledge enrichment such as

WordNet and Wikipedia [13] [14] [80] [5] to improve the categorization accuracy in terms of word and semantic relatedness. Ontology offers capability of knowledge representation in a structural and semantic manner of a specific domain [82] **.** The ontology based semantic annotation and reasoning suggests more for the benefits of the text categorization. In this way semantics and structural information forming the knowledge can be used for various purposes during the categorization process [82] . The utilization of knowledge in the classification of tasks to find associated entities and concepts can be managed using the ontology concepts and relationships. Therefore, the use of ontology has become an intensive topic in text classification [82] .

The aim of research in this thesis is to design and implement a framework that adopt Contextual Ontology based Short Text Messages reasoning, which is able to augment and annotate the short text messages with context ontology for semantic reasoning inference and classification purposes. Basically, the proposed framework adopts context ontology in a given domain with the set of ontological modules that are linked to the text's keywords and metadata. In this method a sequence of terms in the input text are identified and annotated with concepts, entities and knowledge drawn from ontology without the need of learning process.It means that the need to learn pre-classified text for characteristics inferring, which are used in many machine learning methods, are not essential by using ontology concepts. In addition, the proposed framework has the capability of annotating the short text messages categorization after inference process using predefined patterns. The patterns structure is formatted to represent particular ontological module characteristics of domain ontology. In this way, each inferring text messages are tagged to particular pattern, which may help in next matching process for new unknown messages without the needs for ontology navigating.

Various aspects and challenging issues that require extensive research to cope with during the research for providing a realistic and effective approach to handle the limitations are as follows:

- **The description-** Semantic annotation is the basis to realistically approach the semantic reasoning techniques for exploiting the knowledge about information resources. The information resources can be tagged with metadata semantically so that the applications are being well described and effortlessly utilised.

- **Data Modelling -** Ontology is mainly utilised for reasoning and gives additional contextual information. Therefore, it plays a crucial role in facilitating the search and match modules, of which extra knowledge from a query could be inferred to make the retrieval results more efficient and accurate and to help the awareness of message context.

- **Pervasiveness** – Precise data comprehension means better deduction. It is concluded that making decision precise and accurate leads to enable the system to act more pervasively.
- **Intelligence** – Searching extra features and adding further sophisticated techniques to the framework to take intelligent decisions with the help of short messages leverage the application's learning system and hence results in more efficient outcome.

- **Overhead -** The limited resources of low end devices impose constraints to use their capabilities unlimitedly, hence, the minimum data processing and a light weight parsing and reasoning implementation should be investigated for the use in the proposed mechanism.

- **Restriction** – An extensive research is required to address the dependency and uncertainty of message properties, imperfect reasoning and the similarity relatedness threshold rate. These issues should be tackled appropriately to provide an effective running system.

1.3 Contributions to Knowledge

The contributions of this thesis include an automatic semantic annotation of short text messages with entities drawn from a domain contextual ontology without using any type of learning and training. In addition, it supports semantic reasoning based messages awareness for categorization objective by developing a framework to experiment the approach feasibly. It combines features of short text messages categorization with specific approaches to the area of Semantic technology. In particular, the contributions of this thesis are summarised as follows:

- Facilitating mobile communication with automatic short text message annotation by developing the framework (SAMS). The SAMS framework enables annotating key terms of a short text message augmented with an internal and external metadata. The obtained internal metadata is extracted from the header of message plus some of the basic attributes that related to a new composed message from the underlying devices' system. While the external one can be added by the user himself via the framework user interface. The annotation process is accomplished automatically at the time of a message composing as a first step to realise the message context via tagging it with basic metadata attributes. The functional features of XML technology are implemented with the developed framework, in which the aggregating of messages keywords and other metadata are annotated with the composed message to form an XML file simultaneously. A created XML file is of minimum size to avoid the overhead. Thus, it ensures the process of annotation quicker, feasible and unobtrusive as possible.

- Another framework, SOIM, is developed by adopting semantic annotation with context ontology to annotate the text message keywords that have high semantic relatedness with ontology concepts. In addition it improves the semantic reasoning based short text messages inferences and enhances the system capability in terms of short text categorization. The framework SOIM tackles the limitation of crisp annotation and matching with restricted knowledge reasoning of the existing framework- SAMS. SOIM takes the advantage of semantic inference through structuring domain ontology with ontological knowledge modules. The knowledge about a given text message is extracted by computing the semantic distance of message keywords with respect to the ontology. This framework further calculates the aggregated similarities relatedness of keywords to find the best matching category pattern in the ontology schema. The short text message is represented as a set of sub-ontological modules in terms of the proposed domain ontology modules, in which each module with a set of gathered properties and concepts refer to a particular context or event.

- The issue of choosing the best-retrieved pattern that matches input message keywords is tackled by exploiting the keywords' semantic hierarchy and the degree of similarity of the inferred patterns in order to reduce the misclassification error. An enhanced methodology is built on top of the framework SOIM, by adopting rules based Fuzzy Lattice Reasoning (FLR) classifier to deal with the uncertainty of properties. It exploits the hierarchical depth of the short text keyword with respect to the levels of ontology modules to increase the accuracy of the classification process with clearer *d*ecisions.

- The efficiency of the framework performance has been enhanced in terms of time uncertainty and dependency. It was noticed that the message may contain indecisive properties that they do not affect the classifier's decision-making but might affect negatively the search time through the ontology concepts. Therefore, the issue of properties uncertainty is solved using the theory of Rough Sets. The proposed algorithm is proved with experimental results; it is more effective for patterns matchmaking than the SAMS and SOIM. This contribution ensures the system works with less overhead and better performance.

1.4 Research Methodology

The research presented in this thesis was carried out in a number of phases. Initially, in the first stage of my research, a thorough understanding of the issues in the field of study was obtained by investigating some relevant research articles, books, research papers that focused on semantic reasoning, context awareness, mobile computing and its applications. The search was intended to find the state of art about the notion of ambient intelligence environment, and how to adapt it to improve the work of mobile message to render what is called 'Message-Awareness'.

The next phases of the research methodology are summarised as follows:

1. The capabilities of the low-end devices and the relevant programming languages which support such kind of devices were studies in detail. Further, the degree of efficiency of these programming languages and their capabilities of adaptation with limited resources based devices were explored. J2ME as java programming language for limited resources devices was chosen to conduct the framework using the configuration (CLDC) and profile (MIDP).

2. The prototype of the proposed framework, SAMS, structure was detailed. In this stage, the annotation of messages was designed to facilitate the next flow of SAMS stages. The prototype was programmed using JME platform via implementing several MIDlets to perform the annotation, search and matching modules. In addition, the mark-up language, kXML parser was used to conduct the metadata of the annotation process.

3. Performance evaluation tests were assessed to evaluate the efficiency of the prototype in terms of overhead, time consumption and the importance of annotation in this work.

4. Semantic domain ontology has been defined with set of modules, which comprises of a set of similar combined properties and concepts that aggregate in one module to refer to a particular context, event or situation. The ontology leverages the framework performance to enhance the text comprehension. OWL and Jena APIs were used to structure the ontology nodes.

5. SAMS was further improved to the framework SOIM, which employs domain ontology based semantic reasoning. The framework enhanced the search and match modules of SAMS by adopting the semantic distance of the properties in terms of ontology.

6. To proof the SOIM efficiency was measured against SAMS from a number of perspectives. Evaluations, such as computing precision and recall and probability distribution, were accomplished for results validation.

7. The issue of choosing the best-retrieved pattern, that matches input query, has been arisen during the test of the SOIM framework as it affects the decision-making. Therefore, the SOIM is integrated with FLR in order to tune the classification process and enhance the SOIM framework performance.

8. From another side of the framework, during the system learning, a set of predefined patterns were saved in the device for future purposes with a number of properties. Some of those properties may be irrelevant or dependent (indecisive) ones. They decrease the system efficiency by increasing the processing time of matching process; thus, a Rough Sets theory is presented to address the issue of properties uncertainty. This contribution ensures the system works with less overhead and high performance.

9. Experiment results show that using Rough Sets on top of the SOIM is more effective the SAMS and SOIM with respect to patterns matching.

10. Performance evaluation based on pattern searching and matching tests, after integrating Rough Sets in R-SOIM, are conducted and compared against the SAMS and standard SOIM.

1.5 Thesis Structure

Based on the specifications required, a manuscript-based thesis is structured in seven chapters including this chapter (Chapter 1). The organized flow of the chapters helps the reader to investigate the proposed message-awareness framework from its initial structure to the detailed enhancement that was applied to improve its performance in various stages. In the following, a summary of the seven chapters is given, which includes the main contributions that we have made in this thesis:

As introductory chapter, Chapter-1 provides a brief synopsis of the thesis and the fundamental concepts, background, major contributions and the motivation behind this research work.

In Chapter-2, the advantage of semantic web technologies in mobile computing environments has been presented and briefly reviewed. This chapter also discusses the concept of annotation and its significance in the field of information retrieval. First, an overview of the mark-up language, kXML, for the low-end devices has been discussed. The chapter further gives a concise insight about the ontology structure. Further, a brief introduction the Fuzzy Logic and Rough Set theory is presented as well. The chapter also demonstrates some relevant work to the different contributions presented in this thesis.

In Chapter-3, the framework, SAMS, is explained in detail. Here, the annotation and search modules of the proposed framework are elaborated in detail. The chapter presents a detailed analysis of the performance evaluation of the framework with various aspects, specifically concerning its efficiency and accuracy in retrieval and matching. Furthermore, a brief overview of the JME platform and kXML parser, which are employed in message query retrieval based on the use of MIDlets, has been presented.

Based on the previous chapter model, Chapter-4 describes semantic web technologies and their employment techniques in the field of reasoning. It shows the idea of the utilization of ontology and its impact on context and semantic reasoning, particularly in environments where low-end devices are entailed. This chapter also provides a brief introduction to an inference engine and its integration for file searching in the framework. The semantic based file searching framework, SOIM, is presented followed by a comprehensive evaluation of its performance in respect to various measures.

The issue of choosing the best-retrieved pattern that matches input query and its effects on the decision making of the system to follow the right pattern and to do the right action accordingly is elaborated in the Chapter-5. The FLR based rules are proposed and discussed to address the degree of uncertainty to context inference and facilitating the decision-making. Moreover, the chapter presents the reason of how the classifier enhanced the performance of SOIM framework to increase the accuracy of the classification process with clearer decisions. Various aspects of the suggested enhancement are also evaluated in detail.

A fetched query might contains irrelevant and dependents properties, which effects on both the processing time and the efficiency of the framework negatively. In the Chapter-6, the features of Rough Sets are integrated with the framework to empower the awareness process. It focuses on adopting a Rough Sets based semantic retrieval for mobile messages to cope with the properties uncertainty during patterns matching. The proposed framework can retrieve the patterns that are most relevant to a query from a functional point of view. It further filters matched patterns to maximize user satisfaction in message awareness. Set of evaluations with respect to the aspects of accuracy and efficiency are carried out.

Finally, this thesis is summarized in the Chapter-7, in which the research findings of the thesis are concluded and some ideas for the suggested future work are outlined with respect to the research carried out in this thesis.

CHAPTER 2

2 RELEVANT WORK

2.1 Background

In this research, XML and semantic based approaches are considered for studying the awareness of short text messages context. Therefore, before proceeding into the research presented in the following chapters, we briefly discuss both facets of this work. In this chapter, the core concepts and fundamental principles of semantic reasoning technology are presented and highlighted. The notion of semantic reasoning is considered by discussing the concepts of metadata; ontology based semantic annotation, JENA and short text processing in terms of semantic similarity and classification. An overview of the semantic technology and its significance in semantic reasoning and inferring is covered. On top of that, the chapter focuses on reviewing the related literature and prior research relevant to the work on short text categorisation based on external knowledge and data enrichment. Furthermore it discusses the aspects of using an external knowledge represented in the form of semantic ontology, which is proposed in this thesis for short text messages context inference.

2.2 Semantic Technologies

For providing an intelligent knowledge processing, Semantic web technology research is evolving as an extension of the current World Wide Web (WWW). The research mainly aims at to manage the goal of introducing a level of semantics to the available information so that they can become understandable semantically to the machine [113] . The Semantic Web facilitates sharing the explicit semantics of information in a machine readable form [80] [15] . It enables machine to interact efficiently with data and perform many tasks like searching, managing and combining semantically annotated information based on reasoning models. The semantic technology is adopted in various disciplines including Knowledge Management, Context Awareness, Reasoning, Software Agents, *etc*. [113] [15] . In ubiquitous mobile computing, the advancement of Semantic technologies not only enhances the mobile devices capabilities but also enables mobile phones to play a vital role in the daily life due to additional advanced functionalities [32] [33] [34] . The rapid growth in offering diverse services makes the mobile device a key tool for service access in the digital environment [35] .

2.2.1 Metadata

'Data about data' [114] which often refers to the metadata and denotes structured information that describes, explains or manages an information resource to make it easier for retrieval and efficient access to an information resource [36] . Moreover, metadata is the key way to keep the resources accessible easily at any time. Several types of metadata can describe resources at any level of aggregation such [114] :

- Descriptive metadata: It adds the distinguishing descriptions to the information resources identification *e.g.,* keywords, title, summary…*etc.*

- Structural metadata: It is a structural description of the compound objects.

- Administrative metadata: It provides information to help in managing a resource, such as when and how it was created, file type and other technical information and the security as who can access it.

The metadata facilitates the work of Semantic Web technology by using semantic annotation to deduce extra knowledge about resources. Tagging resources with metadata can be semantically exploited in the management and retrieval of text, images, multimedia repositories web documents and file systems [36] . Many techniques have been suggested in developing semantic annotation on different type of information resources. However, very rare work had targeted text messages on mobile phone with respect to semantic reasoning and annotation augmentation. Metadata improves the information retrieval, searching[38] . Different forms of documents such as HTML and Latex are attached with Metadata. In addition, It can be applied to various applications on the web with different kind of languages [115] [114] [38]

2.2.2 Mark-up Language for limited resources devices

XML is a Meta mark-up language that expresses the data exchange and structure across applications and platforms. XML makes plain text form independently which is readable to everyone on different devices. Some XML parsers, which are in charge of XML documents' compilation and manipulation, tend to be bulky with heavy runtime memory requirements [52] [53] . Model, push and pull parsers [117] are mainly the three known types of XML parsers. Model parser reads the whole document and then creates a presentation of it in the memory. However, it requires huge memory depending on the size of the XML document and this memory requirement is more than the other two parsers discussed next. Push parsers process data definitions before the creation of the document as a tree structure in the memory. This process also consumes bulky memory spaces. Thus, this category is not suitable for limited resources devices. The third parser is the pull parser, in which the data is read before parsing process. Its working mechanism depends on dividing the whole document into many parts. Each part is invoked and parsed using recursive functions repeatedly to structure the document tree. Hence, this kind of parser is more suitable for handheld devices as it consumes the devices resources efficiently [54] [53] [52]

Table 2-1: The low light XML parsers

Name	Size	MIDP	Type
ASXMLP 020308	6 kB	yes	push, model
kXML 2.0 alpha	9 kB	yes	Pull
kXML 1.2	16 kB	yes	Pull
MinML 1.7	14 kB	no	Push
NanoXML 1.6.4	10 kB	patch	Model
TinyXML 0.7	12 kB	no	Model
Xparse-J 1.1	6 kB	yes	Model

Many types of XML parsers are developed to interrelate with limited resource devices such as KXML, Min XML, Nano XML and Tiny XML as shown in the Table 2-1 [133] . As a lighter compact version of XML parser, kXML is a kind of pull parser, which is particularly designed for devices that adopt the Mobile Information Device Profile (MIDP) and CLDC configuration, and is exclusively used on the JME platform. Moreover, it can be ported effortlessly to MIDP [51] [50] [49] .

2.2.3 Ontology Web Language

The Ontology Web Language (OWL) is knowledge-representation mark-up language that process information contents besides presenting them to the users. OWL facilitates defining domain ontologies to support the aspects of intelligent pervasive computing [16]

[119] . OWL has the ability to express the semantic of entities better than XML, RDF and RDF-S due its ability to structure specific knowledge in a given domain hierarchically. So that it can be parsed and understood by the machine easily because it can represent machine interpretable content on the Web [118] [119] [2] . Moreover, OWL ontology offers more flexibility in storing, displaying and processing data because it has a normative syntax that can add more vocabularies to describe relations between classes and characteristics of properties[16] . Ontological information can be expressed in OWL language using many predefined classes, subclasses, properties and individuals [16] [118] [119] . A part of the OWL syntax has been used for the same fraction of our proposed domain ontology as shown in the Figure 2-1.

Deriving additional information, truths and statements about modelled ontological knowledge in a given ontology using specific reasoners is one of the main reasons that encouraged developing ontology-based applications. Three kinds of syntax classes are available in the OWL language: OWL Lite, OWL DL and OWL Full. OWL Full is a full version of OWL language which permits more expression and relationships better than OWL DL and OWL Lite. The main objective of OWL Lite and OWL DL is to ease the expression reasoning process. Where, OWL DL can be processed by a description logic reasoner efficiently and a various straightforward inference algorithms can process OWL Lite [136] [118] .

```
<owl:Class rdf:about="mafiles://GhaidOnto/Meeting#lab">
  <owl:sameAs>mafiles://GhaidOnto/Meeting#meetR</owl:sameAs>
  <rdfs:subClassOf>
    <owl:Class rdf:about="mafiles://GhaidOnto/Meeting#office"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="mafiles://GhaidOnto/Meeting#mangm">
  <rdfs:subClassOf>
    <owl:Class rdf:about="mafiles://GhaidOnto/Meeting#business"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="mafiles://GhaidOnto/Meeting#Student">
  <rdfs:subClassOf>
    <owl:Class rdf:about="mafiles://GhaidOnto/Meeting#Formal"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="mafiles://GhaidOnto/Meeting#building">
  <rdfs:subClassOf>
    <owl:Class rdf:about="mafiles://GhaidOnto/Meeting#Place"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="mafiles://GhaidOnto/Meeting#withStf">
  <rdfs:subClassOf rdf:resource="mafiles://GhaidOnto/Meeting#Student"/>
</owl:Class>
<owl:Class rdf:about="mafiles://GhaidOnto/Meeting#ModuleM"/>
<owl:Class rdf:about="mafiles://GhaidOnto/Meeting#house">
  <rdfs:subClassOf rdf:resource="mafiles://GhaidOnto/Meeting#building"/>
</owl:Class>
<owl:Class rdf:about="mafiles://GhaidOnto/Meeting#flat">
  <rdfs:subClassOf rdf:resource="mafiles://GhaidOnto/Meeting#building"/>
</owl:Class>
```

Figure 2-1: A segment of the domain ontology

2.2.4 Jena in Semantic Web

To define knowledge and relationships in a specific domain ontology, a proper storage is required for maintaining and supporting ontology with a specialized query language [82] . Ontology in RDF/S or OWL has a specific logic defined by the choice of the ontology language. A good ontology storage engine must support logic defined by RDF/S or the selected OWL flavor in order to allow correct interpretation of the stored domain knowledge. The used storage strategies vary from a single table of triples to a complex combination of several relational tables that is driven by the ontology schema definition. Jena, Sesame, KAON2 and Redland are among the most commonly used storage systems.

Jena is a toolkit that facilitates semantic web programming in Java programming environment. Jena supports a range of inference engines (reasoners) through its inference APIs to parse and perform semantic reasoning on RDF, RDFS and OWL files to derive additional relationships and information based on Ontology's concepts definitions [57] . Two types of inference reasoning are adopted in Jena which are standard and rule based inference reasoning engines. The standard inference engines such as RDFS and OWL reasoners are built-in reasoners that have been employed instantly by Jena users. Whereas, the rule based inference engines, Jena, gives more flexibility to define new rules using Jena APIs. A rule based engine deduces more relationships than a standard engine and obtains extra statements from fetched data and ontology definitions. User-fetched data follows the format of RDF, RDFS or OWL to be processed in the Jena inference engine. Figure 2-2 shows the Jena infrastructure, which exhibits the Jena reasoning mechanism of generating a new model called an inference model via a combination of both user data and predefined ontology definitions based on model factory [59] [60] [61] . Semantic web technologies are used to develop several context-aware systems based on ontologies, RDF and OWL languages. Context awareness is integrated into the programming model to facilitate dynamic configuration and deployment of mobile services. Jena supports a set of language-neutral ontology APIs, which are independent of the underlying ontology language being used. For instance, each of OWL class, RDFS class or DAML class is represented as the *OntClass* Java class API in Jena. Jena provides a reliable programmatic environment to process information semantically[57]

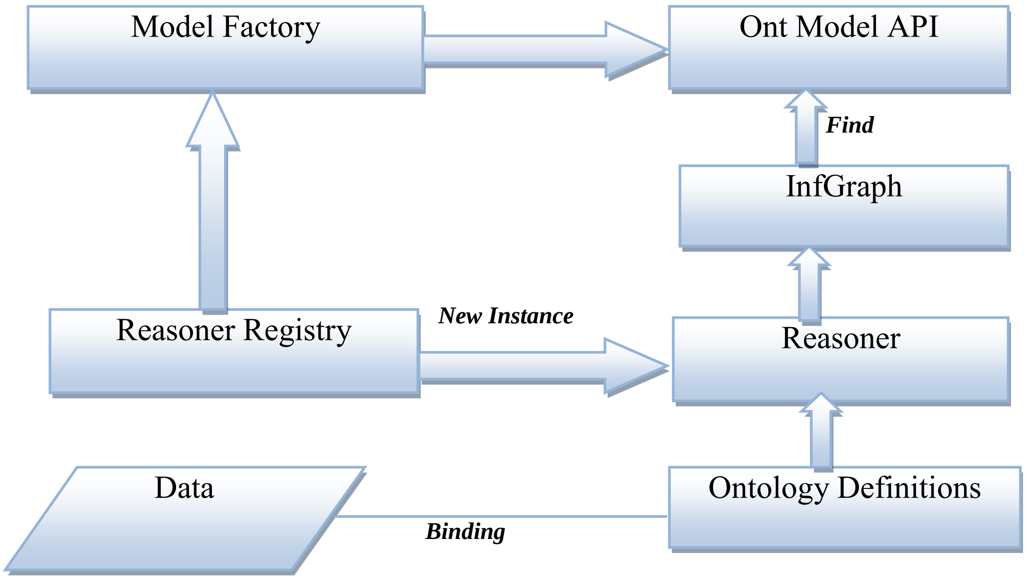

Figure 2-2: Jena mechanism

2.3 Fuzzy Reasoning

Fuzzy Set Theory (FST) handles the concept of gradualness (degrees of membership). It measures the membership of elements in a set using gradual measurement and hence the imprecise context can be represented in a humanly understandable basis [46] [66] . Fuzzy Logic (FL), derived from FST [46] , facilitates decision making based on imprecise, ambiguous or missing context [66] [41] . It enhances the inference capability of fuzzy classifier in learning and applying semantics to classify context. In our case, a Lattice based fuzzy classifier is adopted to enhance the performance of the proposed framework, namely, Fuzzy Lattice Reasoning classifier (FLR) [47] .

2.3.1 General Mathematics Lattice

Mathematically, Lattice L is a partially ordered set that have a greatest lower bound and least upper bound of each two of its elements [138] . A complete lattice L is the set in which its subsets have the least upper bound- $x \wedge y$ (infimum or meet) denoted by and the least upper bound- $x \vee y$ (supremum or join). The inclusion relation can be defined in a Cartesian product lattice $L= L_1 \times \ldots \times L_N$ as [68] [69] :

$$(x_1, \ldots x_N) \leq (y_1, \ldots, y_N)$$

if and only if

$$x_1 \leq y_1, \ldots, x_N \leq y_N$$

The meet and join in L is given by $(x_1,\ldots x_N) \wedge (y_1,\ldots,y_N) = (x_1 \wedge y_1,\ldots, x_N \wedge y_N)$, $(x_1,\ldots, x_N) \vee (y_1,\ldots, y_N) = (x_1 \vee y_1,\ldots,x_N \vee y_N)$ respectively[70] . The flexibility of lattice gives the potential to combine diverse element lattices. It can be used with various types of application such as Fuzzy Set, real numbers vectors, probability space, graphs, etc [70] [47] . It is known that the Fuzzy Set is denote by (U,μ), where U is the universe of discourse and μ is a fuzzy membership function μ: U→[0,1].

The crisp lattice relation '≤' is extended to the notion of fuzzy lattice including the lattice elements that are incomparable satisfying the following definitions [67] [70] :

- A valuation function v: $L \rightarrow R$ in a lattice L is defined by $\boldsymbol{(x) + v(y) = v(x \wedge y) + v(x \vee y), x, y \in L}$. A positive valuation function that satisfies the condition $\boldsymbol{x} < y \Rightarrow v(x) < v(y)$ utilizes an inclusion measure function (σ). the inclusion function is defined on L as σ: $L \times L \rightarrow [0,1]$, such that a set of criteria should be satisfied for any $\boldsymbol{a, b, x \in L}$ [67] [71] [70] :
  - σ(a,O) = 0, a≠ O
  - σ(a,a) = 1, ∀a∈ L
  - a ≤ b ⇒ σ(x,a) ≤ σ(x,b) − Consistency Property
  - a∧b < a ⇒ σ(a,b) < 1. OR y < x∨y ⇒ σ(x,y) < 1.
- Defining the inclusion measure can be represented as: lets <L,μ> is a fuzzy lattice pair, in which L is a crisp lattice and μ: $L \times L \rightarrow [0,1]$ is a fuzzy set membership function , such that $\mu(x,y) = 1$ if and only if $x \leq y$ [67] [70] [71] **.**

A positive valuation function is derived from a real valuation function $v: L \rightarrow R$, which satisfies the rule:

$$v(x \vee y) = v(x) + v(y) - v(x \wedge y), x, y \in L$$

if and only if

$$x < y \Rightarrow v(x) < v(y)$$

It contributes in introducing different inclusion measure functions. Such that, given a lattice L, in which a positive valuation function $v: L \to R$, with $v(O) = 0$, two inclusion measures (2.1), (2.2) can be defined also:

$$f\,(a,b) = v(b)\,/\,v(a \vee b) \tag{2.1}$$

$$y\,(a,b) = v(a \wedge b)\,/\,v(a) \tag{2.2}$$

Where ⟨L,f⟩ and ⟨L,y⟩ are both fuzzy lattices, which can help in solving various decision-making problems [47] .

2.3.2 Fuzzy Lattice Reasoning (FLR)

The notion of FLR classifier is to induce a a rule-based inference engine with set of fuzzy lattice rules ⟨u,c⟩ based on hyperboxes from data that are lattice ordered in a mathematical non-complete lattice data domain including the N-dimensional Euclidean space $\mathfrak{R}^{\mathbb{N}}$. A hyperbox computation in space $\mathfrak{R}^{\mathbb{N}}$ is one of the popular methods of rules induction by assigning a specific point inside hyperbox to its corresponding class. The FLR classifier follows the fast interpretation and computation of hyperbox-based rule induction method which utilizes the advantage of lattice theory to mends the hyperbox disadvantages and improves the classification in a non-complete data set. The FLR classifier work depends on adapting the core area of a hyperbox-shaped of Fuzzy Sets based on the hyperbox diagonal with minimal elongation [47] [67] .

A fuzzy lattice rule is presented as a pair $\langle l,c \rangle$ where l is an element in a fuzzy lattice $\langle L,\mu \rangle$, $l \in L$ and c is a class label that refer a set of predefined classes $c \in C$. The objective of inducing fuzzy lattice rule is to implement a function $f: L \to C$, which maps an element l to a class c ($l \to c$) according to a rule 'If antecedent then consequent'. The antecedent is a set of conjunctive fuzzy expressions corresponding to inputs and the consequent part is a class label [46] [67] .

The FLR classifier is expected to a map f: $\langle U,\mu \rangle \to C$ according to A fuzzy lattice rule engine $E\langle L,\mu\rangle$, which is a set of fuzzy lattice rules $\{a_i \rightarrow c_i\text{: } a_i \in \langle L,\mu\rangle\}$. The reasoning of FLR implies the computation of the degree of truth for every rule to match 'a' with the proper class 'c'. For example, Let x be an input element, x_1, x_2, x_3 are fuzzy lattice elements with rules engine $E\langle L,\sigma\rangle = \{x_1 \rightarrow c_1, x_2 \rightarrow c_2, x_3 \rightarrow c_3\}$ [71] and σ a fuzzy membership function defined in the Equation (2.1). In an iterative comparison process, the truth of the following consequence: $c_1 = \sigma(x, x_1)$, $c_2 = \sigma(x, x_2)$, and $c_3 = \sigma(x, x_3)$ are calculated, in which the element antecedent $x \in L$, of unknown class label, is presented over the rules $x_n \rightarrow c_n$ of the engine. The decision making of similarity assigns x to the category c_J, where [71] [47]

$$J = arg \max_{l \in \{1,\ldots,R\}} \sigma(Q \leq P_l) \tag{2.3}$$

2.4 Rough Sets Theory

Rough Sets theory is "an extension of conventional set theory that supports approximations in decision-making" [139] . It is an approach to vagueness set, in which a boundary region of a given set can be used to express its imprecision, a contrast to Fuzzy Set theory, where a partial membership is adopted.

Rough Sets concept is characterized by using operations, *interior* and *closure*, called approximations[140] . Namely, lower and upper approximations, 'are a classification of the domain of interest into disjoint categories' [73] [77] . Figure 2-3 shows the upper and lower approximation is presented along with the edge between them.

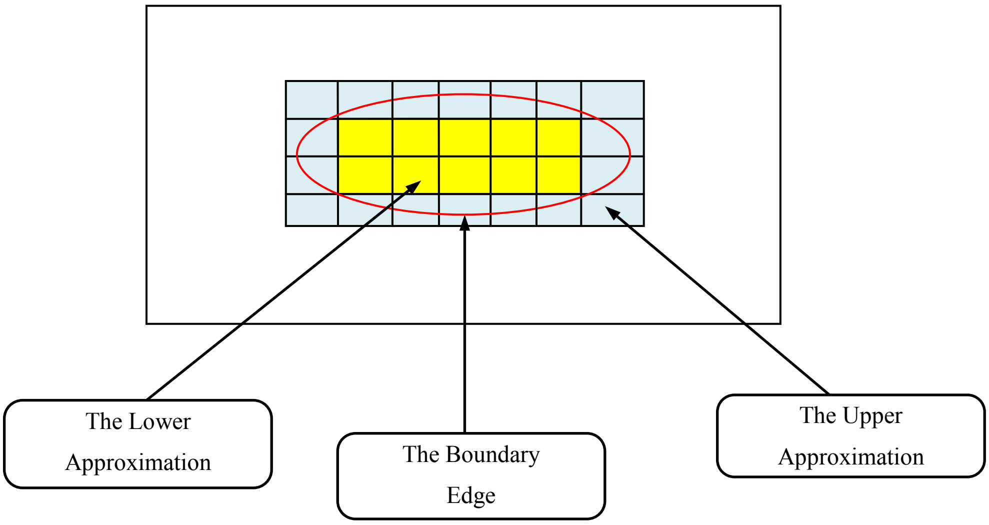

Figure 2-3: Approximation in Rough Sets [77]

A fundamental principle of Rough sets as a mathematical concept is to extract filtered knowledge from a given domain. In addition, it facilitates eliminating all redundancies and dependencies in a given set's features [140] byretaining the significant information only and reducing the irrelevant involved knowledge [77] [76] [73] .

Let I = (U, A) be an information system, where U is a non-empty set of finite objects , A is a non-empty finite set of attributes such that $a{:}U \rightarrow V_a$ for every $a \in A$. Indiscernibility relation *IND*(*P*) is an equivalence relation that express the lack of knowledge to discern some elements in the universe.[77] [73] :

$$IND(P) = \{(x,y) \in U \times U \mid \forall a \in P \;\; a(x) = a(y)\} \qquad (2.4)$$

It is used to define approximations concepts of rough set as follows [74] [76] [77] :

- *P-lower approximation* of *X*

$$\underline{P}(x) = \cup_{x \subset U}\{P(x): P(x) \subseteq X\} \qquad (2.5)$$

- *R-upper approximation* of *X*

$$\overline{P}(x) = \cup_{x \subset U}\{P(x): P(x) \cap X \neq \phi\} \qquad (2.6)$$

- *P-boundary region* of *X*

$$P_B(x) = \overline{P}(X) - \underline{P}(X) \quad (2.7)$$

In which, the *lower approximation* of a set *X* with respect to *P* is union of all granules objects, which are classified in certain way as *X* with respect to *P*. While the *upper approximation* of a set *X* with respect to *P* is union of all granules, which have non-empty intersection with the set; in which the objects can be *possibly* classified as *X* with regard to *P*. Moreover, the difference between the upper and the lower approximation result in the boundary region of objects' set, in which its objects can be classified neither as *lower approximation set* nor as upper approximation set in terms of *P*. Two rules are derived to distinguish a set X with regard to the particular aspects of the rough set theory concept [62] [78] [77] :

- Set *X* is *crisp* (precise), if the boundary region of *X* is empty $\boldsymbol{P_B(x) = \phi}$
- Set *X* is *rough* (imprecise), if the boundary region of *X* is nonempty $\boldsymbol{P_B(x) \neq \phi}$.

Rough set is also considered by computing *accuracy of approximation* as following [77] [73] :

$$\alpha_P(X) = \frac{|\underline{P}(X)|}{|\overline{P}(X)|}$$

Where $|X|$ stand for the cardinality of X and $0 \leq \alpha_P(X) \leq 1$. If $\alpha_P(X) = 1$, X is *crisp* (*X* is *precise* in terms of *P*), and otherwise, if $\alpha_P(X) < 1$, *X* is *rough* (*X* is *vague* in terms of *P*). Indescernibility relation *IND*(P) partition the universe U into a set of granules of knowledge with respect to *P*, which is computed as follows:

$$U/P = \otimes\{a \in P: U/IND(\{a\})\} \quad (2.8)$$

$$A \otimes B = \{X \cap Y: \forall X \in A,: \forall Y \in B, X \cap Y \neq \emptyset\} \quad (2.9)$$

So the positive, negative and boundary regions can be represented in (2.10) (2.11) (2.12) respectively, where P and Q be equivalence relations over U. The *positive region* $POS_P(Q)$ contains all objects of U that can be grouped to clusters of U/Q with respect to P [77] [73] [74] .

$$POS_P(Q) = \cup_{X \in U/Q} \underline{P}(X) \tag{2.10}$$

$$NEG_P(Q) = U - \cup_{X \in U/Q} \overline{P}(X) \tag{2.11}$$

$$BND_P(Q) = \cup_{X \in U/Q} \overline{P}(X) - \cup_{X \in U/Q} \underline{P}(X) \tag{2.12}$$

Employing rough membership function is also acceptable to define the Rough sets as [73] [77] :

$$\mu_X^P(x) = \frac{|X \cap P(x)|}{|P(x)|}$$

Where, $\mu_X^P : U \rightarrow < 0,1 >$ and $|X|$ is the cardinality of X. Figure 2-4 depicts the formation of rough membership function, which illustrates to what extent x belongs to X given P [77] [73] .

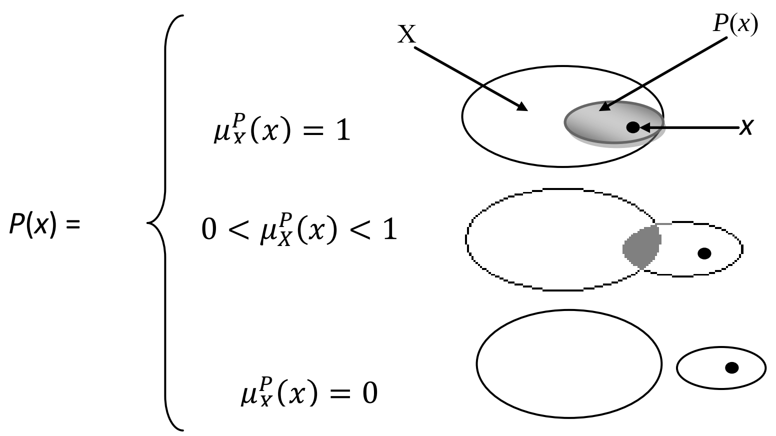

Figure 2-4: Rough Sets membership function

Rough Set theory was applied to reduce the uncertainty of pattern properties when matching set of selected patterns with message query by eliminate any irrelevant indecisive properties which are dispensable in matchmaking. The objectivity of Rough theory sets has been approved in data processing as it does not need any preliminary or additional information about data compared with approaches that are based on Dempster Shafer theory and Bayesian networks [131] [132] .

2.5 Related work

2.5.1 Semantic Annotation

'Content-based reasoning' is one of the key challenges in the field of Text documents' semantic awareness managements system as inference new knowledge requires automated text reasoning. This can be achieved via developing approaches of automated semantic annotation, in which a text document is annotated with ontology concepts[10] and the facility of information exchange between ontology and text documents [10]. One potential benefit of automated annotation is increasing the scalability needed for text document annotation on the Web. In addition, it can reduce the load of annotating new documents and applying knowledge based ontologies [121] . Web application, such as Semantic Web, is facilitated using the fundamental technology of text annotation with ontologies knowledge and concepts [10] The initial set of manual annotations are initiated to be attached to text documents and then automated annotation system can suggest extra Meta information to the user to extend the text's metadata [26][10] In [24], the authors have adopted an approach to map text headings to ontology's entries. However, the mapping is based on exact matching between a specific ontology concept and the title of a text fragment using transformations methods such as N-grams and stemming algorithm to performance improvement. A new method is proposed in [23], in which data extraction ontologies for specific domain are utilised to annotate Web pages using automated semantic annotation. In spite of the notion that adopted to avoid the techniques of extracting information heuristics, in this research annotating candidate instances with concepts of a given domain ontology require an expert of that domain in order to import its formalised semantics. Linguistic patterns that express semantic meaning of annotated text documents with named entities are implemented by [22] where the proposed mechanism selects the best pattern that match to the annotated entity. Although the accuracy of this method is sufficiently high, its recall is limited as only named entities are annotated, which exist in specified documents in the Web pages. Similarly, Ontea system, in [25], has adopted Web documents annotation based on lemmatization methods and regular expression patterns. The method limitation here is the need for predefined patterns for specific domain is required which affects the system performance.

In our work, we add to this flow of work the speciality that the input texts to be annotated are composed of very limited terms, namely, mobile short text messages. A short text message is annotated automatically with domain ontology concepts through semantic annotation without using any type of learning and training. Where, the key idea is to identify a sequence of terms in the input short text and to annotate them with entities drawn from domain context ontology. Specifically, the research in this thesis adopting a novel approach of automated annotation based short text messages reasoning.

2.5.2 External Knowledge based Data Enrichment

Due to the rapid emergence of short text applications such as mobile text messages, search snippets, product reviews and a summary of a text book, the need for short text processing is revealed [92] . Many traditional techniques for document similarity measurement or classification that rely on word frequency are usually effective with long text documents as they mostly contain many co-occurring words, hence; it represents the text as a Bag Of Words (BOW) which focuses on word frequency and word co-occurrences [7] [91] . However, the limited content and contextual information, poor grammars and sparse nature have an effect on techniques like standard NLP and traditional similarity measures to attain desired results and detect intrinsic relationship among short text snippets. In addition, due to the data sparseness, achieving reasonable accuracy has become a challenge in short text classification. Evidently, the limited words in short text makes the detection of intrinsic relationship among text snippets difficult using traditional similarity measures. Several considerable issues are entailed with Short text classification such as:

- Classification methods have become a challenge in the research area because of the Short text messages data sparseness. Hence, additional incorporated methods are required for short text messages inflating [6] .

- Adapting semantic similarity between entities is important in terms of time and domains as the lack of content and context affects a semantic similarity accurate computing. For example, in spite of 'apple' is frequently allied to computers on the Web search, most general-purpose thesauri are still not listed that [92] .

- Desired results require to classify and inference short text precisely which cannot be achieved by using the Standard NLP techniques because of the conciseness of Short text and the lack of its regular grammar [6] .

Currently, the short texts classification and reasoning has been tackled in the literature by inflating a short text with meta-information from external sources to make it appear like a large document of text via deriving new features from external knowledge bases, such as WordNet and Wikipedia.

External knowledge can improve short text processing by enriching original text with related information, strengthen co-occurrences and it can help also in vocabularies unifying [93] [94] . In WordNet, it is a network of semantically related words, groups of words, and phrases. WordNet was originally created as a combination of a dictionary and a thesaurus. It is used mostly to unify the vocabulary across the documents by modifying the document features with the use of the related words. WordNet is successfully used in document clustering presented in [95] to leverage the semantics of words and in the categorization algorithm described in [93] . External knowledge may be used to collapse some terms into a common class or a term to strengthen the selected features. For example the terms 'lamb' and 'beef' are similar, as they are both defined as sub concepts of the word 'meat' in WordNet. In addition, Word sense disambiguation has been solved in WorldNet as in [95] , where synsets are used to disambiguate sense for different terms in the analyzed document. The authors in [96] have used WordNet in word sense disambiguation and synonym expansion to measure sentences similarity via providing a richer semantic context.

Recently, various studies have searched the possibility of adopting Wikipedia resources for various text categorization and NLP tasks [82] [98] Some of text categorization approaches utilize Wikipedia over multiple domains as it contains a vast amount of general knowledge with very detailed descriptions that is interconnected and categorized. Wikipedia can be also used to tackle a *polysemy* (the meaning ambiguity) problem in natural language processing, where it is used as a source of knowledge about the semantically related neighbor entities [99] . A research study in [100] has used Wikipedia, as a background data to extract a number of hidden topics in articles and gaining more knowledge for user defined categories. In this research, the extracted hidden topics are firmed by the specific dataset instead of the external knowledge base.

Some initial work has been done to leverage the Wikipedia structure with the Semantic Web knowledge representation paradigms by conversion the Wikipedia structure from an XML document structure into an RDF/S, so that it can work as an ontology for the categorization purposes [101] and produce a semantic Wikipedia [101] . Both WordNet and

Wikipedia have been applied by Hu *et al* [103] to enhance existing features, in which WordNet was used for keywords and Wikipedia was used for concepts enrichment.

This means that the need to go further beyond the related words to the knowledge utilization, associated entities and concepts are significant step to short text inference and classification. Ontology offers knowledge of a given domain that is organized both in a structural and semantic way, as it provides named entities and relationships between them. In addition, term disambiguation and vocabulary unification can be successfully solved by ontology. The authors in [93] proposed using synonymous words, multi-word phrases and polysemous words to overcome deficiencies when the used features are only word-based. Ontology can describe relationships among named entities as well as helps to recognize them. It can help to discover multi-word entities in the document which are treated as separate words. Also, disambiguated named entities can be assigned to multiple contexts. In [104] , the co-occurrence of certain pairs of words or entities have been reinforced with the ontology term vectors but are not clearly related in the document corpora. Further utilization of the ontological knowledge is presented in [106] and [105] .

Recognizing a proper context with the help of ontological entity classification and other entities present in the document can efficiently narrow down the target categorization domain and improves accuracy. Text categorization with the help of a dynamically obtained ontology is described in [107] . The extraction of text features and keywords from the training corpus is used to create a map of associated terms; hence, they are transformed to ontology and association rules to enhance the standard categorization process. Automatic text categorization can also benefit from using encyclopedic knowledge that is encoded in the form of ontology.

A successful use of ontological features and the encyclopedic knowledge in text reasoning and categorization is an important indication that ontology-based semantic inference and categorization on short text messages may be possible. In addition, it is worth further investigation as the well-defined external knowledge can improve the quality of knowledge deduction.

2.6 Text Categorization

Text Classification is a supervised data mining technique that involves assigning a document to one (or more) classes, based on its content, *e.g,* classifying emails and messages such as Spam, Work *etc.* Many aspects determine the efficiency of a good text classifier such as categorizing large sets of texts in a reasonable time with acceptable accuracy and classification speed.

Basically, a classifier adopts a particular learning method that is applied to a training set of categories and then assigning the categories labels to label new unknown objects depending on gained knowledge during training phase of the classification process. Text dataset of text documents with their class labels are fetched to the classifier during the training stage to enable the system generate set of rules based on a given learning method [143] . Different kinds of algorithms and learning methods have been developed to leverage the aspect of automatic text categorization in the literature. For instance, decision trees, naive-Bayes and rule induction that have been applied to solve set of classification problems and achieved satisfactory results [108] . However, the current automated text categorization classifiers are not faultless and needs improvement. Further, the time to train a classifier is significant and therefore, the effectiveness of classification is still a rich area of research. Majority of the existing algorithms for text categorization can be classified as supervised, unsupervised and semi-supervised methods. Supervised learning aims to learn the knowledge from the already labelled training data and then apply this on the testing data and predict the class label for the test data set accurately. Some supervised methods include probabilistic Bayesian models and nearest neighbor's classifier. While the Semi-supervised method tries to learn from labeled data and enhance the classification function with unlabeled examples.

The notion of learning is how to deal with the issues of decision-making and knowledge imprecision. Significant research efforts have been dedicated to capture, infer and classify the context [46] [41] . Some research have been done on context inference, however, they have not tackled the semantic reasoning extensively. For example, in [42] , the authors have proposed the Dynamic Bayesian Network where a context inference's model has been proposed without the semantic capabilities. Consequently, logic based probabilistic classification for unknown context deduction is adopted in [43] . However, it has not utilized hierarchy semantics. The authors in [44] have investigated the situations representation using core ontologies and therefore have d enhanced semantics and knowledge reasoning.

Fuzzy logic has been implemented by many researchers to enhance learning and allow a degree of uncertainty during the context estimation and decision-making. In [45] , a context inference, represented through Fuzzy Sets, is used for detecting unknown context which is derived from multisensory environments. The suggested work has not taken into account the hierarchy semantics. Anagnostopoulos *et al*, in [41] has adopted a Fuzzy Set to handle reasoning imperfection with vague situations and uncertain decisions by learning an appropriate threshold for decision making. They have applied the rule (**If** *premise*, **Then** *consequent*), where *premise* is a part of fuzzy linguistic values that represent the current situation. Furthermore, relying on hierarchy semantic (specialization, mereological compatibility) among context, the authors in [46] have used Fuzzy learning to deduce the imperfection of context.

In this thesis, we have used A Fuzzy Lattice Reasoning (FLR) [67] to estimate the best retrieved pattern with high similarity degree that match a new unknown text message. The capability of FLR of using hyperbox with latticed order supports our methods as a classification process which depends on the properties hierarchal depth in the domain ontology. Therefore, an ordered set of properties can be generated as a fetched dataset to the classifier. The FLR has shown good results as compared with the three other classifiers which have been chosen to test the same dataset.

2.7 Summary

Most of the related research work on short text messages has primarily tackled the data sparseness problem by enlarging the text with extra information using external knowledge bases such as Wikipedia, WordNet. Evidently etc. These techniques have achieved better results in terms of accuracy against classical techniques. However, increasing the feature set leads to 'curse of dimensionality' problem [38]. Similarly, querying the online knowledge base poses new problems with real applications where longer time is required. In this work, we propose the use of ontological modules as an external knowledge base in which each module refers to a specific event or context with common feature set to inferring short text messages for classification purposes. Initially, diverse pre-defined patterns are defined with respect to the modules of ontology such as festivals, events and schedules. This method deals with semantic domain ontology as a third party to compare between the new unknown short text message and the module pattern to be categorized.

CHAPTER 3

3 FACILITATING MOBILE COMMUNICATION WITH ANNOTATED MESSAGES

3.1 Overview

In this chapter a framework, namely (SAMS) - facilitating mobile communication with short text annotated messages [133] is proposed and implemented**.** It facilitates annotating key terms of a short text message augmented with an internal and external metadata, in which the annotation process is carried out automatically at the time of composing a message. The functional features of XML technology are implemented with the developed framework in which both the messages' keywords and other metadata are annotated simultaneously with the created message as an XML file. The SAMS framework keeps the created XML file with the minimum size to avoid the overhead. Besides, it saves message metadata locally, and hence, requires neither common repository nor communication medium for metadata storing and retrieving. Thus, it ensures the process of annotation as quicker, feasible, and unobtrusive as possible.

This chapter discusses the proposed framework components: annotation, search and matching modules. Two user cases studies have been presented to test its working process and effectiveness. The framework has been evaluated in terms of parsing time and overhead as well as the importance of metadata in improving the message classification.

3.2 The SAMS Framework

Some mobile phones' short text messages content can be attached to specific user situations, *e.g.,* a message context with set of gathered keywords (properties) forms a context of a particular event. For instance, a message text related to holding a meeting. This means that it is more likely the text will contain terms or keywords related to the meeting context like meeting location, date, time, and even its purpose. Situation awareness applications concentrate on a current user situation to enable a device adapting to this situation and trigger proper actions accordingly. Evidently, in some cases involving a message content as a part of the situation context helps the system to make decision more accurately, *e.g,* a user is in a business meeting and he is not supposed to be disturbed receiving messages from others except the ones related with his work. In this case, depending on only the user location, profile or calendar does not help the system to trigger the right action when a new business message is received. In another example, sometimes we want to receive messages from only a specific contact number in a particular time and date. Therefore, understanding the semantic of text messages to be a part of the user current situation may leverage the devices' context awareness and enhance a decision–making accuracy. Annotation is the first step to realise the message context via tagging it with basic metadata attributes. The framework SAMS aims at to annotate a short text message with part of speech tags in addition to internal and external metadata that is entered by the user at the time of message creation. The significance of annotation process in this research focuses on identifying the tagged message keywords with domain contextual ontology (which will be detailed in the Chapter 4). The candidate keywords with high degree of similarity will be annotated with text message and hence performing semantic inference process. In the following sections, the SAMS framework structure, which is illustrated in the Figure 3-1, is discussed in detail.

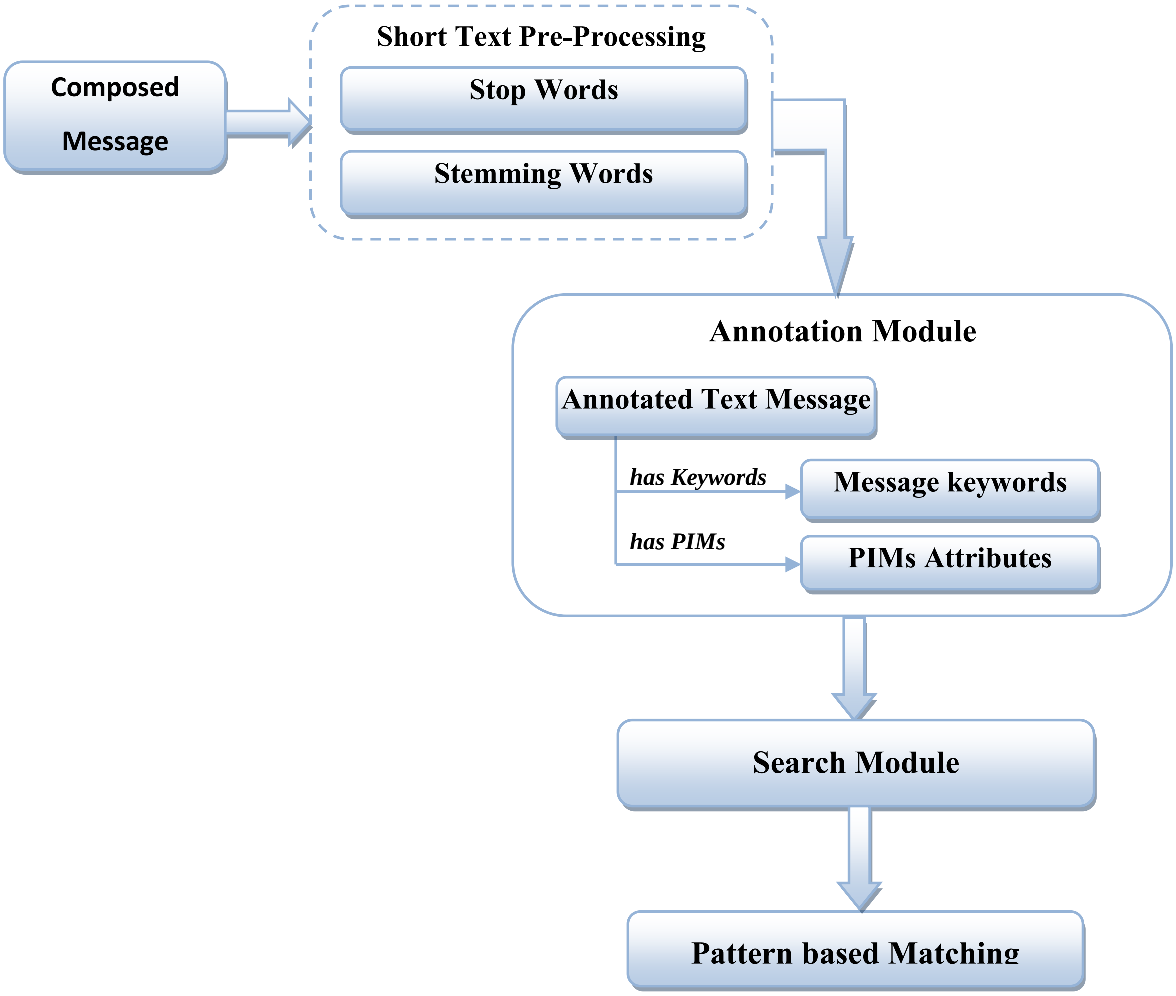

Figure 3-1: The SAMS framework structure

3.3 The SAMS Implementation

The SAMS comprises of three consecutive components: short text pre-processing, annotation module and search/match module.

- **Short text pre-processing**

  Some common words are usually not considered in search engines in order to speed up the processing and hence the search results. These filtered words are known as 'Stop Words' (such as is, are, the... etc) [89] . Stemming is a technique to find morphological variants of search terms for improving information retrieval performance [20] . It refers to the act of conflating or combining the variant, hence, this reduces all words with the same root to a single form and increase the recall accordingly. Stemming can be processed manually or programmatically using computer program called stemmers such as Porter [20] .

- **Annotation module**

  The SAMS framework aims at to annotate a text message with its keywords excluding any irrelevant words such as part of speech and some other basic attributes. The purpose of annotation process is to augment messages with metadata in an efficient formation in order to facilitate them during search and retrieval. Section 3.4 discusses the short text messages' annotation process in the SAMS framework in detail.

- **Search module**

  In this module, when annotated message is received, its extracted metadata from parsing stage is matched to the saved patterns where SAMS can search for a specific corresponding pattern that matches the text message tagged keywords. In detail, after parsing process, the whole message metadata is extracted using kXML parser, the search module searches the required pattern based on extracted metadata. Extracted metadata in each message are message keywords, two basic attributes and an optional user category tags. Search can be accomplished using extracted attributes. Two ways of searching the SAMS can be performed. The first one, when a user selects a message category, in this case SAMS searches the patterns via navigating all the patterns belonging to the specified category. Each message's keyword is compared to pattern's properties and retrieves a degree of match between them. The search in other way navigates through all patterns' categories as the user's option tag is left empty. When the degree of match is calculated to all saved patterns against message keywords, the pattern with exact match is retrieved only. The crisp matching retrieval due to the SAMS framework's search module does not adopt any enhanced semantics with restricted knowledge reasoning.

3.4 Message Annotation

Annotation module interrelates with composed message; the Figure 3-2 shows the process diagram that explains the structure of annotation module. Basically, annotation module in the framework automatically annotates the text message with corresponding underlying vital attributes that are saved in the message header and device's file system on the device. Then the metadata is stored locally and augmented with the message keywords after pre-processing. The system attributes are extracted using the PIM APIs for Java environment, which is a package, defined in the Java Specification Request (JSR)-75 [55]

The basic attributes are extracted from the message header such as an addressee phone number, the date of a composed message. Afterwards, they are annotated automatically at the time of message creation. Moreover, the framework allows the user to tag the message with a message subject (message category), which is optional selection, *i.e.*, it is not obligated to tag a created message with its subject. The SAMS framework is represented via a user interface that includes: the addressee phone number, the message subject and the message as shown in the Figure 3-3, which illustrates the main user interface.

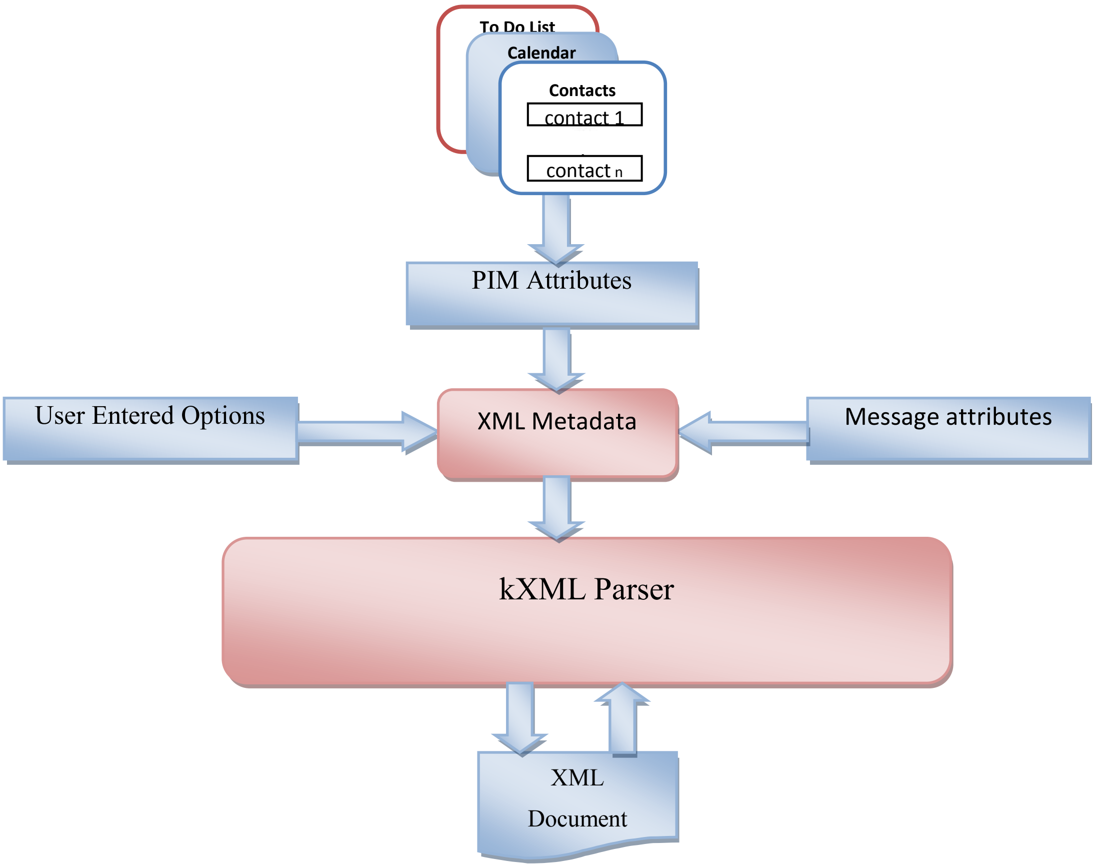

Figure 3-2: Annotation module

Suppose a given message query M_p with a set of N properties M_p={p_1,p_2, ….., p_N} and A_t ={a_1,a_2,….,a_l} are the attributes extracted from the system and/or user. The metadata aggregated from both $M_p(p_i)$ and $A_t(a_j)$, according to the following rule syntax:

$$\forall \boldsymbol{p} \in \boldsymbol{M_p}, \boldsymbol{a} \in \boldsymbol{A_t} | \boldsymbol{Ant}(\boldsymbol{M_p}(\boldsymbol{p}) \wedge \boldsymbol{A_t}(\boldsymbol{a}))$$

Where p, a is a message keyword and an attribute extracted from the device system that is related to the composed message, *e.g.*, addressee phone number respectively. $Ant(M_p(p) \wedge A_t(a))$ is an aggregating operation, in which both of the p_i and a_j unite together as tagged metadata attributes in an Xml format. Therefore, each message can be annotated with three attributes along with the text message keywords as shown in the Figure 3-4, which depicts a fragment of the stored metadata represented as an XML file. The created XML file is parsed using MIDlet based kXML parser. The extracted attributes of parsing process can be utilized in the search module of the framework, where XML data are navigated for searching a required match pattern that match an incoming message query.

Figure 3-3: SAMS main interface

```
<? Xml version='1.0' encoding='ISO-8859-1'?>
<Annotated data>
<Msg Content> where the room of the Business meeting today </MsgContent>
< has Keyword1> room </ has Keyword1>
< has Keyword2>Bussiness < has Keyword2>
< has Keyword3> meeting < has Keyword3>
< has Keyword4>today< has Keyword4>
<hasCategory> Meeting Schedule </ hasCategory >
<hasPhone No.>  07506267001 </ hasPhone No.>
<Date> 24/05/11 </Date>
</Annotated data>
```

Figure 3-4: Annotated metadata in XML format code

3.5 Case Studies

Two scenarios are presented in this section to reinforce a full understanding of the functional principle of the SAMS framework with the significance of proposed work in message awareness, which is the core objective of this research.

3.5.1 Address Book Management

In this scenario, a user should not be interrupted by receiving messages while he is busy (*e.g.,* in attending important meeting). However, if his partner sends him message enquiring information urgently, *e.g.,* acquiring contact information - beforehand assuming that a message sender is endorsed in his authorized group list for security purposes, the device next starts acting to the existing context according to the user's current situation. Basically, in this case, a contact query properties –XML code- as shown in the Figure 3-5(a-d) is sent to parsing module (*i.e.,* the kXML parser) to extract the information from the message tags. Then, SAMS searches for query patterns in terms of tags' information (as aforementioned in search module), when search is finished, the device can infer the query category. Accordingly, two specified actions are assumed to be triggered by the device, which are "take action" or "take no action". Taking action can be triggered only when the text message query keywords are matched to the retrieved pattern completely.

-a-

-b-

-c- -d-

Figure 3-5: SAMS, address book management,

(a) Text query, (b) The annotated message in XML format, (c) text inferring (d) assumed action

In this case study, the framework capability to annotate, process, act to a text message are investigated by assuming a desired action taken by the device without further user involvement is carried out. This means, the device navigates the phone address book and loading all the personalized information related to the contact list that is intended for comparison process. If the demanded query about specific contact is found during the search process, the system will automatically terminate the search process and retrieve the detailed information about that contact. Consequently, the obtained information is passed on to the action module to react and activate an Auto reply action and re-forward the requested information back to the query-sender device. Otherwise, the second selection "take no action" denotes that if the device is uncertain about the current context to act to a query, thus, it does not invoke any activity/task if a certain event occurs (*e.g.*, query properties do not match any predefined set of patterns on the device).

3.5.2 Schedule Management

Meeting schedule is another scenario that is verified to test the framework -SAMS. Suppose a user wants to arrange for a group meeting, an SMS message is supposed to be circulated to all involved members with the meeting details *e.g.*, the date/time, the location and the purpose of meeting. Basically, for confirming a schedule slot, everyone has to access his phone calendar to confirm his availability based on date and time slot of the proposed meeting schedule. In this scenario, SAMS derives the idea from the aforementioned by simulating an automatic scheduler, of which the calendar resource of mobile phone can be controlled and managed autonomously

The core work here concentrates on reaching the user's phone calendar and checking the slot availability of the requested meeting query against all other reserved slots. After publishing a text message query to connected group, SAMS on each member's device follows the ordinary trail of the framework modules, starting from parsing process until performing an appropriate action, as elaborated in the previous case study. The search module manages the retrieved data from the kXML parser by accessing the device's calendar resource to check the availability of requested date and time in the message query by using JSR75, PIM APIs. In the case of availability, the system would adjust the calendar slots corresponding to demanded query. Subsequently, confirmation alert would be sent back to other connected nodes on the network as demonstrated in the Figure 3-6 (a-d). Otherwise, no action would be carried out and the control will be transferred to the handheld device user to deal with the query manually.

-a- -b- -c- -d-

Figure 3-6: Meeting schedule user case:

(a) Sending date, (b) The received XML document, (c) text inferring (d) assumed action

3.6 Performance Evaluation

To reveal the practicality of the SAMS implementation, tests and experiments were conducted to evaluate its performance from the aspects of efficiency and time processing of annotation and matching processes.

3.6.1 Parsing performance Evaluation

Time consumption by the SAMS framework is divided between the parsing time accomplished throughout the process of kXML parser and search time during matching process. The size of the created XML file is not static as it depends on query text tags. However, it is not enlarged significantly because of the text message's size is and tags' options have limited content. According to different implementation on various sizes of XML files, it is shown that an XML file of one kilobyte size needs approximately 4.07 milliseconds. Therefore, the conclusion is that the kXML parser takes no longer time in metadata parsing process. Furthermore, another test was implemented to evaluate the search time of the proposed framework by applying 5 independent tests. Each test encompasses a diverse set of patterns ranging from (20 to 250) with different sizes as shown in the Table 3-1. It is apparent from the results that the SAMS performance is not degraded because of the spending time in searching process *e.g.*, a set of 70 patterns take about 79 milliseconds with reference to the searching time might varies with respect to the size of patterns knowledge base and the first successful match.

Table 3-1: Searching time evaluation

number of patterns	Time taken (in ms)
25	19
70	79
150	146
250	286

3.6.2 Geometric Distribution

In probability theory and statistics, Geometric distribution refers to the probability of the number of independent trials required to obtain the first success and desired result as given in the Equation (3.1) [134] .

$$G(x,p) = pq^{x-1} \tag{3.1}$$

Where p and q the probability of success and failure respectively, G is a Geometric distributed variable, and $x=1,2,\ldots,n$ gives the number of n trials in which the first successes occurs.

The efficiency evaluation of the framework is assessed in terms of the annotation process's significance, particularly the optional metadata – a message category – in supporting the pattern retrieval acceleration. Two sets of tests were formulated, each comprising of 50 text messages. The first set includes fully annotated messages, whereas the second set embraces messages, in which the user's selection tag is left empty. The success rate of 68% and 49% for first set and second set, respectively, was obtained.

Let p_a, p_o and q_a, q_o are the success and failure probabilities for the fully annotated and partial annotated sets respectively, G_a , G_o, is the geometric distribution for the probability of x_{ath} , x_{oth} trial that shown the first successful retrieval for both sets. Using maximum likelihood estimator, the geometric distribution results based on (3.1) of the both sets are shown in the Table 3-2. Obviously, from the table results, the probability value of p_a, p_o is 0.68 and 0.49 respectively, needs a value of x_a = 1.47 and x_o =2.04 trails to get the first successful required pattern.

Table 3-2: Geometric distribution

rail's No /x	1	2	3	4	5	6	7	8	9	10
Ga	.68	.2176	.0696	.0223	.0071	.0023	.0007	.0002	.0001	.0000
Go	.49	.2499	.1274	.0650	.0331	.0169	.0086	.0044	.0022	.0011

The value of success probability p plays an important role in the geometric distribution results, of which a clear perception about the framework performance in both cases is observed. So that, the framework evaluation based on full-annotated attributes has the higher chances for getting the first successful query retrieval with a small number of trials in comparison with missing annotated attributes as shown in the Figure 3-7, which shows the graph of varying the probability of the first search success with respect to number of trails.

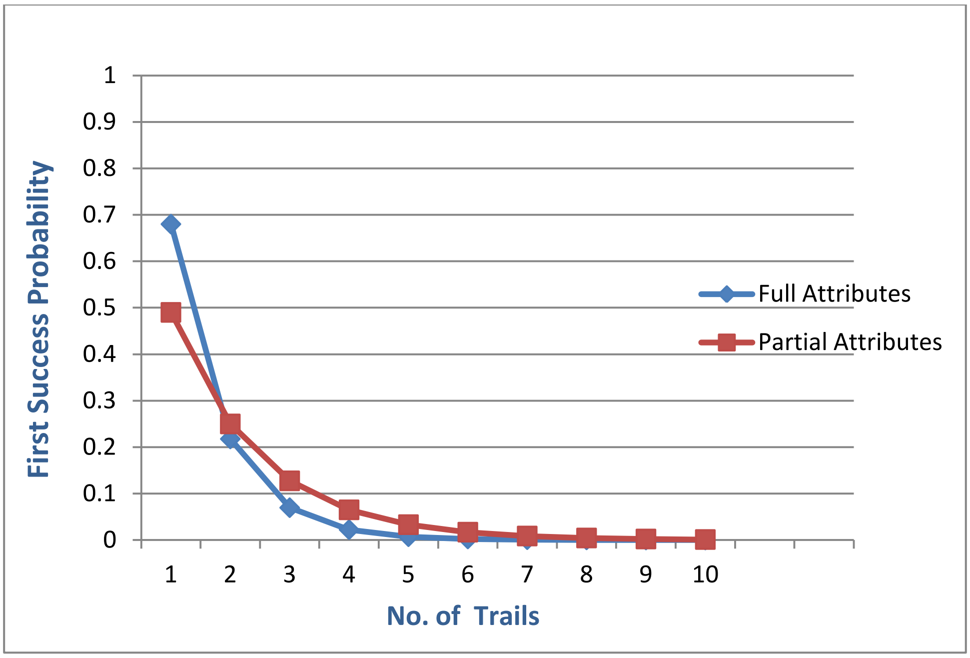

Figure 3-7: The success probabilities (Geometric distribution)

3.7 Summary

This chapter presented the framework of SAMS structure in which the mobile messages are annotated with metadata on low-end devices and specifically focusing on the utilization of XML structure in message management. Furthermore, it provides an efficient mechanism to search message query and retrieve the required action associated with the requested query in search module. It is based on two main modules: annotation module and search module, which are highlighted in the framework. Based on various aspects and analyses, SAMS performance has been evaluated. However, the drawback of this framework is the lack of enhanced semantics with restricted knowledge reasoning. Therefore, semantic based enhancements of the framework are proposed in the next chapters.

CHAPTER 4

4 ONTOLOGY BASED SHORT TEXT MESSAGES REASONING

4.1 Overview

An ambient smart world is achieved via integration of devices into the user environments intelligently. Specifically, context aware computing is emerging as one of the trails to achieve a goal, in which services are seamlessly available at right time, right place and in the right situations. This chapter presents a framework, SOIM, which adopts semantic inferences of short text messages based on contextual domain ontology [55] . The proposed framework further enhances the search capabilities of SAMS, as presented in the Chapter-3, via semantic reasoning inference capabilities enrichment. SOIM facilitates short text message annotation and semantic reasoning capabilities with domain ontology.

The framework structure is built on semantic technologies and supports automatic processing of short text messages on low-end mobile devices. It adopts message inference based on ontological knowledge modules to improve the system capability in terms of message categorization and leverage the situation awareness. Implementing semantic inference with ontology extracts specific knowledge about a given text message by computing the semantic distance of message keywords with respect to the ontology. It further calculates the aggregated similarity relatedness of keywords to find the best matching category pattern in the ontology schema.

The JENA framework APIs that defined on Java mobile platform is implemented to carry out the SOIM prototype. The performance of the SOIM framework is evaluated from a number of aspects, and experimental results are encouraging showing the effectiveness of SOIM.

4.2 SOIM– Ontology based Messages Reasoning

Essentially, the framework SOIM relies on the hierarchical semantic distance to compute an approximated match degree of new set of relevant keywords to their corresponding abstract class in the domain ontology. Applications and services in realistic pervasive computing environments are usually clustered as a collection of sub-domains, in which each domain shares common concepts and knowledge such as health, home and work. So that they can be modeled using a general context model but with different detailed features [135] . Context modeling can facilitate abstract concepts reuse and defining application-specific knowledge.

Domain ontology is modeled into set of ontological knowledge modules with different levels that capture features of contextual entities and features. It means a text message is treated as set of contextual information that refers to a particular situation or event. For instance, a meeting schedule module may contain four abstract concepts, namely- meeting category, meeting location, meeting date and time. These concepts may have much more details and sub-concepts, and hence, the SOIM framework takes into account the semantic level of matched keywords when computing the similarity. The SOIM has initiated a set of predefined patterns that are derived from the ontology modules with different levels to form categorized clusters that share common properties and may be connected to a similar action that might be taken by the user/device in a specified context. For example, let a new text message query related with medical appointment " ***Do not miss the appointment with your doctor tomorrow morning at GP*** " is published, hence, set of patterns is initiated to this published query such as:

[(<*category*> *medical meeting*, <*place*> *GP centre*, <*time*>*morning*),
(<*category*> *hospital appointment*, <*place*> *medical centre*, <*time*>*day*),
(<*category*> *formal meeting*, <*place*> *building*, <*time*>*Time*)].

As can be seen from the example above, the patterns can be structured based on the module's hierarchical structure. Then, a similarity relatedness of the aggregating text keywords is calculated with respect to these retrieved patterns.

4.3 The Design of SOIM

The SOIM framework components as shown in the Figure 4-1 interrelate through two processes – System publication and system matching.

- **System Publication.**

  Basically, the composed text message is annotated with metadata before publishing (step 1). Domain ontology can be defined in Ontology Web Language (OWL) [88] [57] . This OWL ontology is then parsed by an OWL-full reasoner, which it is then loaded into the ontology repository (step 2). When structuring ontology with modules, initiated patterns are set based on these modules with different levels (step 3) and then saved them in the patterns repository (step 4), in which the patterns are categorised and attached to specific actions based on their elements.

➢ **System Matching**

Firstly, when annotated message is received, its extracted metadata from parsing stage is passed to inference engine (JENA is used in our work) to infer additional knowledge and statements about the text (step 5) in terms of domain ontology. The ontology repository is used by an inference engine to infer the semantic relationships of properties during matching text message query (step 6). Then upon completion of knowledge inferring, the search module is invoked to match between the message queries to the patterns with respect to the knowledge extracted from the ontology (step 7). Subsequently, the similarity degrees of all matched patterns to the text query using the match degrees of their individual properties are computed. Finally, list of retrieved patterns that match the new text message query are ranked with their similarity degree (step 8). The pattern match with highest degree of similarity is chosen to be a categorical reference to the new unknown text message, which may be saved in pattern repository for next search and matching (step 9).

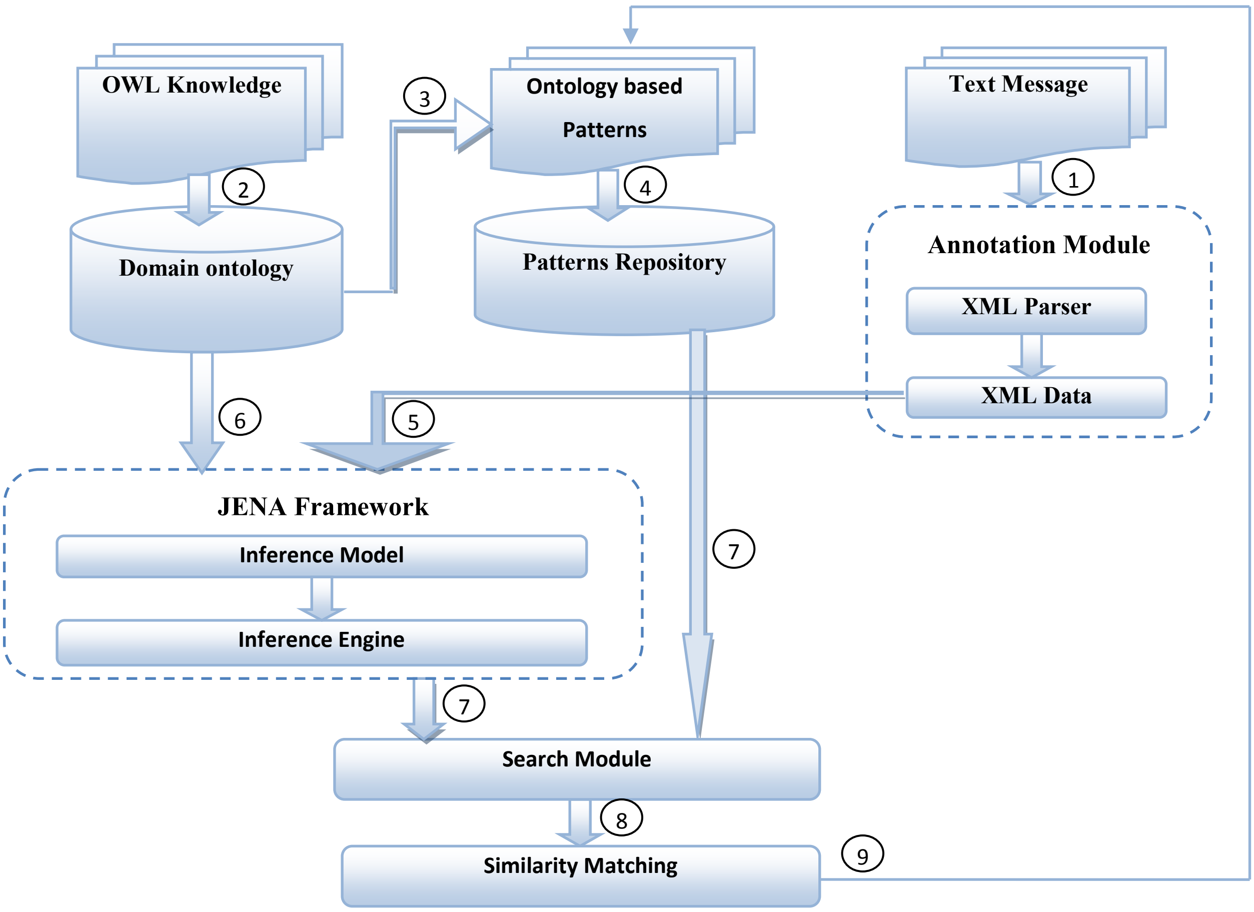

Figure 4-1: SOIM Framework Architecture

4.4 The Semantic Reasoning of SOIM

Annotation module in the SOIM framework is followed by the same procedure of the proposed and implemented one for the SAMS framework as explained in the Section 3.4, Chapter-3. However, the SOIM framework differs from SAMS in searching mechanism because it adopts the semantic reasoning technology to enhance the framework capability of search, match, categorization and retrieval functions. The annotation procedure comprises of a semantic matching, in which the calculation of relatedness between tagged message keywords and ontology concepts is computed to provide much flexibility in words reasoning compared to the exact matching that accomplished in the SAMS framework. SOIM is built on top of JENA framework to facilitate mobile messages with annotation and discovery using semantic inferences of a short text query.

4.4.1 Context Representation

Contextual information of the structured ontology in the SOIM framework can be represented as a set of subontological modules$O_i \,|O_i \in O, i = 1,2,\ldots,m$, where O is the structured domain ontology, and *m* is the number of modules in *O*. Each subontology O_i comprises a set of concepts $\mathrm{C}(O_i)$, properties $\mathrm{P}(O_i)$, and individuals $\mathrm{I}(O_i)$ with axioms to declare the concepts and properties, and facts to declare the individuals. Subconcepts are defined as concepts' branches representing more specific information based on the axiom $c \sqsubseteq o;\ c, o \in \mathrm{C}(O_i)$, in which *c* is a subconcept subsumed by *o*, (e.g. *o*=Time, *c*= Morning). The description of property *p*(*x*,*y*) in terms of ontology follows the rule :

$$p(x, y) \in P(Oi)|x, y \ \in I(Oi); c(x), c(y) \in C(Oi)$$

Where, *x* and *y* are instances of a concept *c* (e.g. *meeting room* is an instance of *place* concept – *Place*(*meeting room*)), and *p* is the property that links up the individual *x* with *y* (e.g. *Locate*(*seminar*, *lab*)). An OWL ontology reasoner is used to infer additional knowledge from O_i to leverage matching process. The reasoner infers individuals, subconcepts and extra statements from the main concepts to compose further connections among them. User message query properties $Q(q_i)$ are classified as a type of a predefined event $P(p_j)$, $P \in O$ when all or parts of its attributes are matched and harmonized with the selected pattern in terms of ontology definitions. Nevertheless, imprecise knowledge may lead to an incomplete partial matching between input query and the patterns with respect to ontology. Thus, semantic distances based a degree of similarity between q_i and p_j is involved in matching process.

4.4.2 Search Module

The search module in the SOIM works as follows:

Lets $Q(q_i)$ be i^{th} attribute of a message query Q and R represents a solution space of the set of predefined patterns as follows:

$$S = \{S \subset R, q \in Q, r \in R \mid Q(q) \rightarrow R(r)\}$$

Where S is the solution set that is deduced from R with a specified measure of match between q and r, in terms of their location in O. As illustrated in the Figure 4-1, the structure of SOIM comprises the components: patterns knowledge base, domain ontology and text discovery service. The repository of patterns hold an initial set of predefined patterns (historical records of patterns) that are extracted from the ontology modules levels, in which each inferring text messages are tagged to a particular abstracted pattern. Besides, the saved patterns may help the subsequent exact matching processes of new unknown messages without the need for ontology navigating (the SAMS framework scheme). The domain ontology repository contains general properties and keywords about specified contextual events. An OWL reasoner parses the domain ontology to infer more semantic relationships among properties that will be utilized when matching a published user message query to the patterns.

The message query attributes are explored through navigating the inference model with respect to the ontology properties. Then, the inferred query properties are forwarded to the patterns repository to deduce and identify the relevant task patterns to the query properties. Upon completion, both the query and task patterns are passed to the text message discovery service (the match module) in order to calculate the degree of match of the pattern properties against the query properties. However, a match degree of zero is retrieved for each an irrelevant property. Consequently, list of patterns that are ranked with their matched degrees are retrieved. The similarity degree of the selected pattern properties to the whole set of message query properties is computed using the summation of the match degrees of their individual properties. As a result, SOIM retrieves the patterns with the highest similarity degree. Because of the limited computing resources of low-end devices such as mobile phones, it is unlikely to perform the reasoning process in rational time. Therefore, the XML file with the meta-data of the message query is sent to a server, in which the reasoning is performed by binding it with ontology definitions. The XML file is processed effectively on a resource limited device using the kXML parser because it is particularly designed as a light version of an XML parser for low-end devices and it is supported in the JME platform (CLDC and MIDP)[93][144] .

4.4.3 Similarity Matching

To match message properties to the retrieved patterns properties in terms of the ontology definitions, a numerical match degree is computed as:

Let

- p_Q be a property of the received text message query
- p_A be a property associated with one of the saved patterns.

Based on the work done by [62] , which computed match degrees to quantify the relationships of properties in an service advertisement and service query by considering the semantic distance between them in terms of OWL ontology for service discovery purposes [62] , the following relationships between p_Q and p_A are defined as :

- *Exact match:* p_Q and p_A are equivalent, or p_Q is a subclass of p_A.
- *Plug-in match:* p_A subsumes p_Q.
- *Subsume match:* p_Q subsumes p_A.
- *Nomatch:* There is no subsumption between p_Q and p_A.

A semantic distance between the text message query p_Q and retrieved matched patterns p_A is calculated by using the degree of match Equation (4.1), which assigns a numerical degree for each match to quantify the relationships between p_Q and p_A as defined above . Hence, to consider the semantic distance between p_Q and p_A , Let:

- $||P_Q, P_A||$ be the semantic distance between p_Q and p_A in terms of the domain ontology O.

- *dom*(p_Q,p_A) be the degree of similarity between p_Q and p_A

Then

$$dom(P_Q,P_A)=\begin{cases} 1 & exact\ match \\ \frac{1}{2}+\frac{1}{e^{(\|PQ,PA\|-1)}} & plugin\text{match}, \|PQ,PA\| \geq 2, \\ \frac{1}{2\times e^{(\|PQ,PA\|-1)}} & subsume\text{match}, \|PQ,PA\| \geq 1, \\ 0.5 & uncertain\text{match}, \\ 0 & no\ match \end{cases} \qquad (4.1)$$

If *dom*(p_Q, p_A) ∈ (0.5, 1), a plug-in match will be produced between p_Q and p_A. Else if *dom*(p_Q, p_A) ∈ (0, 0.5), a subsumed match will produced be between p_Q and p_A.

4.5 The Framework Evaluation

In this section, different tests and comparisons are outlined to evaluate the SOIM framework in terms of the similarity relatedness accuracy and retrieved matched.

4.5.1 Similarity Relatedness

A suggested scenario is developed to investigate and explore the SOIM implementation. Suppose that Ami has received a text query to attend a staff meeting. Ami is unable to interact with the text directly, thus, SOIM on her device is activated to process the incoming text as illustrated in the Figure 4-2. Following the SOIM process, a number of patterns (see the Figure 4-3 a segment of schedule module ontology) is retrieved, *e.g.*,

[(*<category> meeting,<profile> business manager <place> conflab, <time>day*), (*<category> business <place> meeting room, <time>weekday, < category> Internal meeting*), (*<location>* Indoors, *<category> formal meeting, <place> building, <profile>contact number*)].

(a) **(b)**

Figure 4-2: Query processing, (a) text message query sending from node ***A***, (b) text message processing in node ***B***

According to the aforementioned scenario, a test is performed on a set of random queries patterns that are selected in order to compute their degree of match to two types of queries. 30 patterns were selected from patterns repository with at least one property has a valid relationship (*exact*, *plug-in*, or *subsume*) with one of the query properties as mentioned in the aforementioned section. Two forms of query properties were selected to evaluate the match degree. In the first form of the query, all its properties were relevant and defined clearly in terms of the domain ontology (see the Figure 4-3), whereas in the second one, at least half of its properties are irrelevant. The first query is ***"meeting Wed 11:00 conflab"*** and the second one is **"*this evening at 18:00 a seminar will be held in LecRoom*"**, respectively. For the second query, the properties with a match degree of zero were not returned. Based on the Equation (4.1), the match degree is calculated as illustrated in the Table 4-1 and 4-2 for query (1) and (2) respectively.

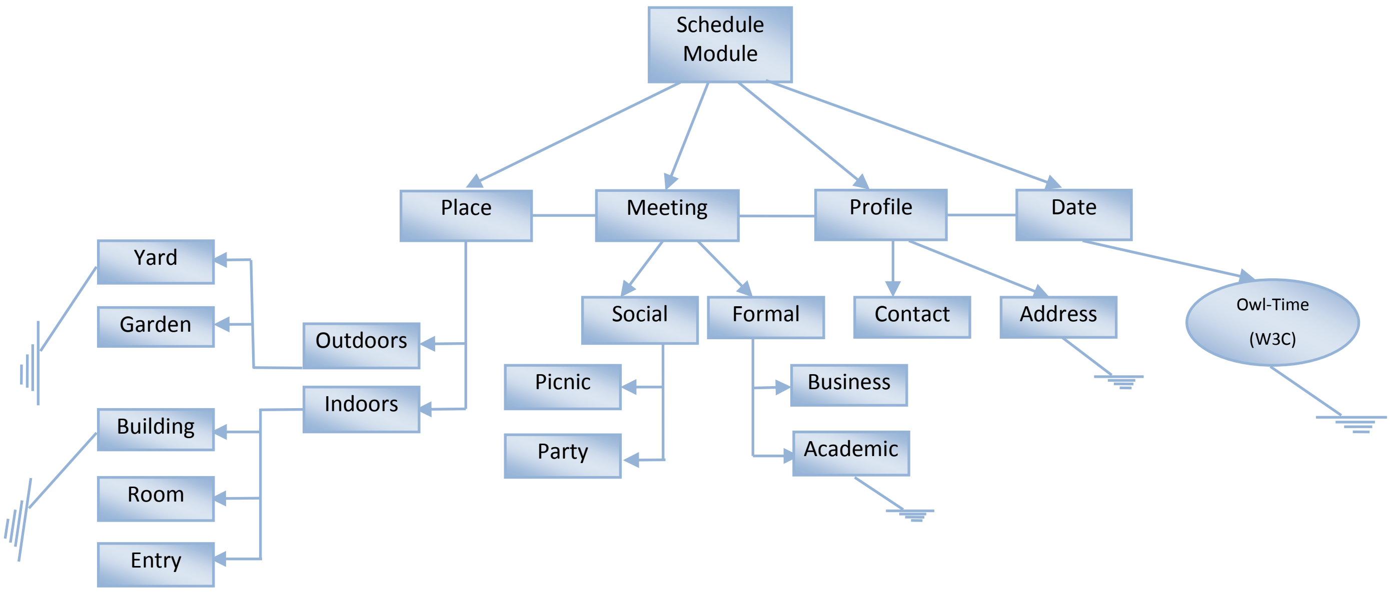

Figure 4-3: A module of domain ontology

Table 4-1: Match degrees for query-1

Properties / *Retrieved Patterns*		***P1***	***P2***	***P3***	***P4***
SQ1	*<meeting, date, hour-minutes(Indv), conflab>*	E	U	P	E
SQ2	*<business appointment, weekday, time ,meeting room>*	P	S	U	P
SQ3	*<Academic meeting ,indatetime, duration, instituation>*	S	P	S	P
SQ4	*<meeting, dayofyear, timezone, conflab>*	E	U	S	E
SQ5	*<meeting, weekday(Indv), hour-minutes(Indv), conflab>*	E	E	E	E
SQ6	*<formal meeting, days, intervalstart,conflab>*	P	P	S	E
E: Exact (100%) - P: PlugIn (50% or more) - S: Subsumed (50% or less) – U: Uncertain (50%)					

A degree of similarity is computed using the Equation (4.2) [62] , which is the mean value of the maximal match degrees of every property of a selected pattern when all the properties used in the query.

$$S(Q,s) = \sum_{i=1}^{M}\sum_{j=1}^{N} \max\left(dom(PQj, PAi)\right)/M \qquad (4.2)$$

Table 4-2: Match degrees for query-2

Retrieved Patterns \ Properties		P1	P2	P3	P4
SQ1	*<indatetime, hour-minutes, lecture, lecroom>*	S	E	P	E
SQ2	*< intervalstartby, timezone, seminar, lecroom>*	P	S	E	E
SQ3	*< intervalstartby, hours-minutes, academic, room>*	P	E	S	U
SQ4	*<intervalduring, intervalbefore, seminar, lecroom>*	E	U	E	E
SQ5	*<intervalcontain, intervalstart,formal meeting, institution>*	U	P	S	U
SQ6	*<indatetime,hasbegining, seminar, lecroom>*	U	P	E	E
E: Exact (100%) - P: PlugIn (50% and more) - S: Subsumed (50% and less) - U: Uncertain (50%)					

Where p_Q, p_A are a set of N, M properties used in the query and the saved patterns respectively, and $dom(p_{Qj},p_{Ai})$ is a match degree between p_{Qj} and p_{Ai} in terms of ontology O. As shown in Table 4-3, the similarity degree (applied on test2) using the SOIM framework performance is better in compared with the SAMS framework. The reason is that SAMS deals with an exact match only, thus, in S_{Q5} the S_D is 0% because no property was assigned to exact match.

Table 4-3: The similarity degree of SOIM vs. SAMS

Retrieved Patterns \ Similarity Degree	SAMS	SOIM
SQ1	50%	79%
SQ2	50%	78%
SQ3	25%	70%
SQ4	75%	81%
SQ5	0%	56%
SQ6	50%	74%

4.5.2 Precision and Recall

To evaluate the accuracy of the search mechanism precision in SOIM compared with SAMS, Precision and Recall has been used as standard measures to assess the information retrieval[63] . Recall (*Rec*) and precision (*Pres*) can be calculated as follows:

$$Rc = \frac{|\mathrm{Re}trel|}{|\mathrm{Re}l|} \qquad PREc = \frac{|\mathrm{Re}trel|}{|\mathrm{Re}t|}$$

Where *Rel* refers to relevant patterns, *Ret* be the number of returned patterns and the returned relevant patterns is denoted as *RetRel*.

Precision and recall were calculated by selecting 50 different patterns randomly for three times using 20 relevant patterns and at least 4 properties were accompanied with each pattern. Two sets of tests were conducted on the 50 patterns. In the first set of tests, at least one pattern was selected that matches the query. A selected pattern had at least one property assigned to an *exact* match to guarantee that both SAMS and SOIM return all the relevant patterns. No constraints were enforced on the second set of tests in order to show the effectiveness of SOIM with semantics. The results of precision and recall were calculated on test 1 and test 2 by computing the average of all the three times of calculations. It is observed from the Figure 4-4 and Figure 4-5 that SOIM achieves better performance than SAMS because of the latter is based on exact keyword matching while SOIM mainly depends on semantic inference techniques. The precision of SOIM is always higher than of SAMS.

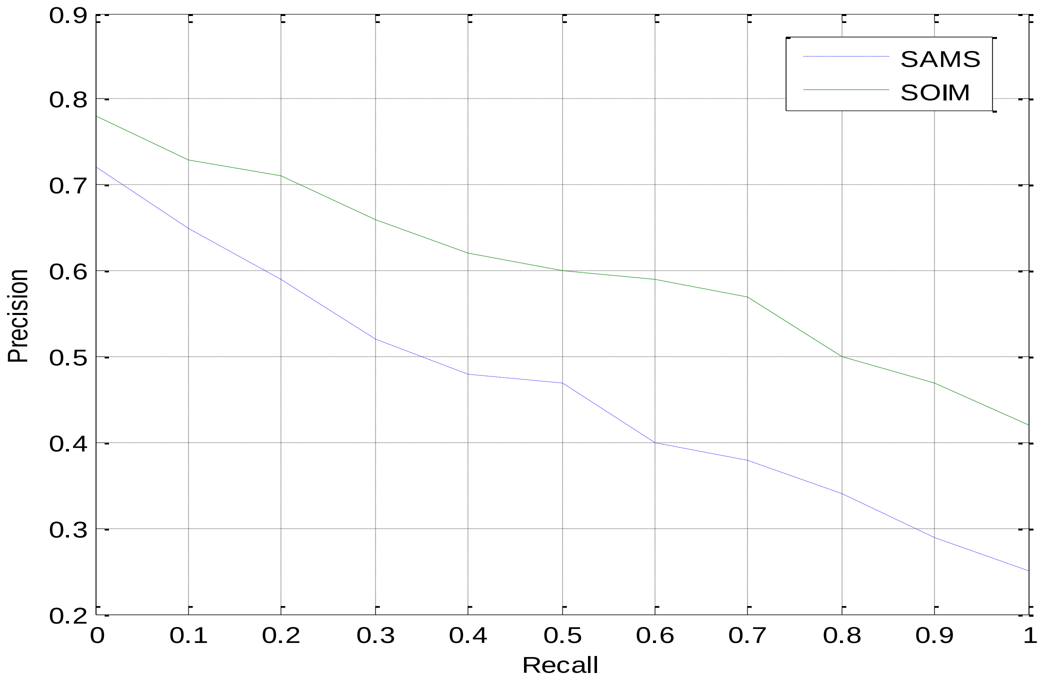

Figure 4-4: Precision and Recall (test 1)

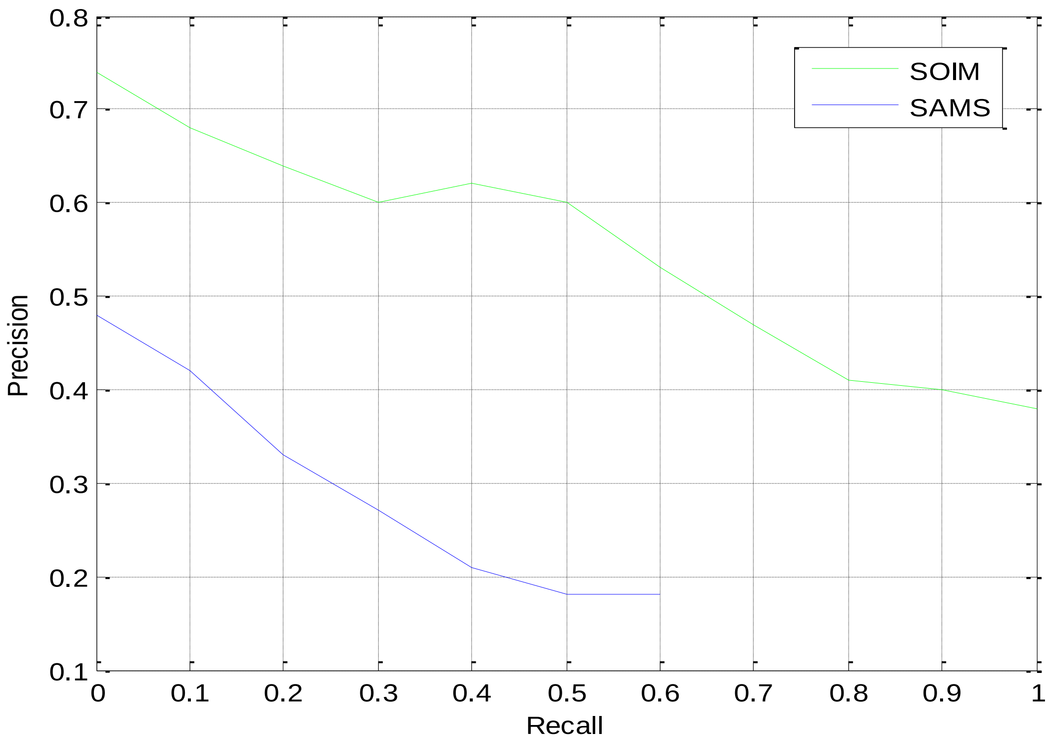

Figure 4-5: Precision and Recall (test 2)

4.5.3 Geometric Distribution

To observe the framework performance with respect of the success probability, the Geometric distribution had been carried out based on the Equation (3.1) in the Section 3.6.2 in the Chapter 3. The efficiency evaluation of the SOIM framework is assessed and compared to SAMS framework to calculate the probability of retrieving the first success patterns. We have chosen 120 patterns for test, half of them are fully annotated while the other half has no information about the query category, *i.e.*, the category tag is left empty. Three tests have been performed, the first test applied on the set of patterns with missing category tag value, the second test was applied on the set with full annotated attributes, while the third test was conducted on the both sets altogether.

Let

- p_{sa} , q_{sa} , x_{sa} ,G_{sa} represent the success probability, failure probability, the trial that shown the first successful retrieval in SAMS.
- p_{so} , q_{so} , x_{so} ,G_{so} represent the success probability, failure probability, the trial that shown the first successful retrieval in SOIM.

Table 4-4: The retrieved Patterns of the three tests

Tests Number	Number of Retrieved Patterns -SAMS	Number of Retrieved Patterns -SOIM	p_{sa} SAMS	p_{so} SOIM
1	54	84	0.45	0.70
2	62	86	0.52	0.72
3	61	91	0.51	0.76

The retrieved patterns of both the frameworks with respect to the three conducted tests are shown in the Table 4-4. The successes rate of the tests is shown in the Tables 4-5, 4-6 and 4-7 respectively. These tables present the geometric distribution calculated for 10 trials of each framework. As seen from the plotted Figures 4-6, 4-7 and 4-8, the first success probability of SOIM is higher as compared to SAMS in spite of the values were close in all three tests.

Table 4-5: Geometric distribution - test 1

Trail's No	1	2	3	4	5	6	7	8	9	10
Gsa	0.45	0.2475	0.1361	0.0749	0.0412	0.0226	0.0125	0.0069	0.0038	0.0021
Gso	0.70	0.2100	0.0630	0.0189	0.0057	0.0017	0.0005	0.0002	0.0000	0.0000

Table 4-6: Geometric distribution - test 2

Trail's No	1	2	3	4	5	6	7	8	9	10
Gsa	0.52	0.2496	0.1198	0.0575	0.0276	0.0132	0.0064	0.0031	0.0015	0.0007
Gso	0.72	0.2016	0.0564	0.0158	0.0044	0.0012	0.0003	0.0001	0.0000	0.0000

Table 4-7: Geometric distribution - test 3

Trail's No	1	2	3	4	5	6	7	8	9	10
Gsa	0.51	0.2499	0.1225	0.0600	0.0294	0.0144	0.0071	0.0035	0.0017	0.0008
Gso	0.76	0.1824	0.0438	0.0105	0.0025	0.0006	0.0001	0.0000	0.0000	0.0000

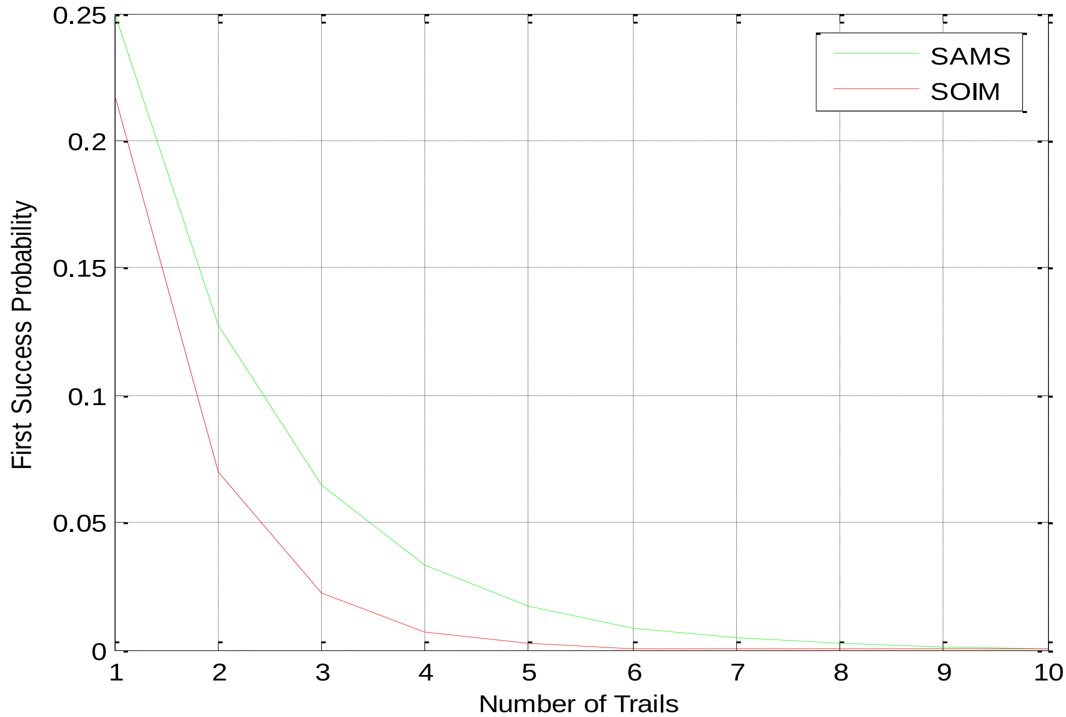

Figure 4-6: The success probabilities (Geometric distribution)-test 1

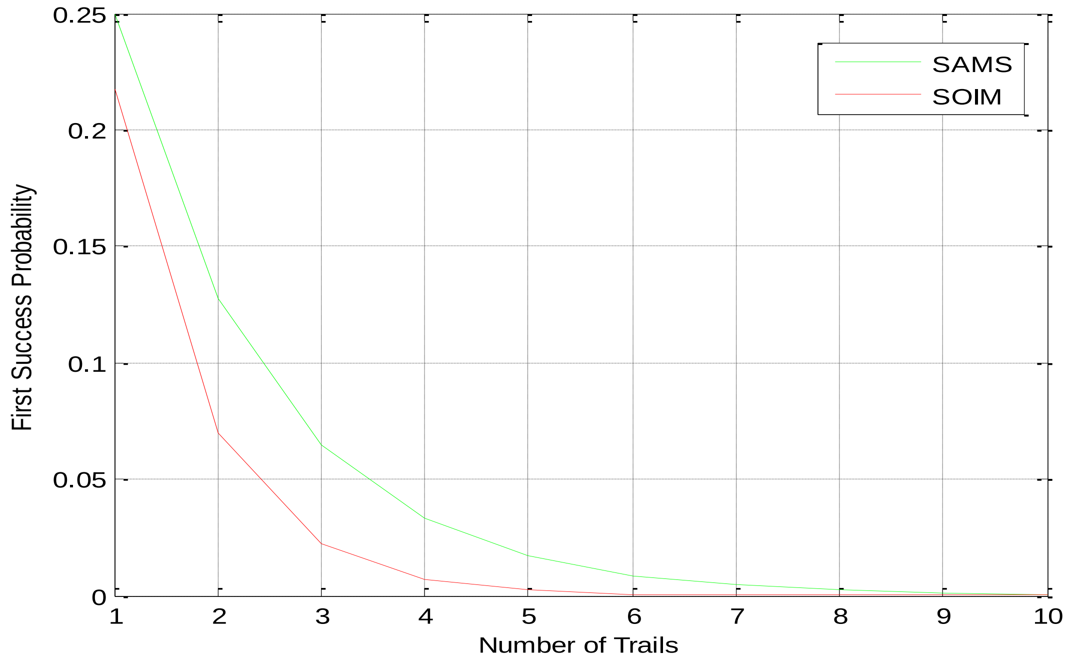

Figure 4-7: The success probabilities - test 2

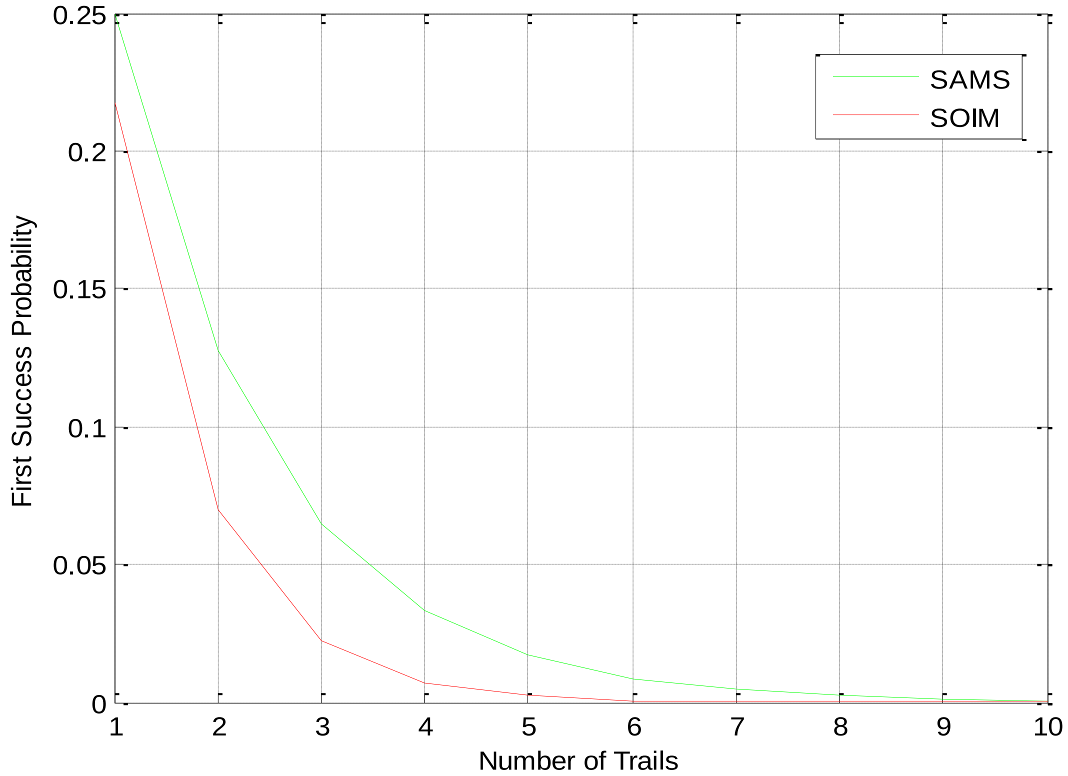

Figure 4-8: The success probabilities - test 3

4.6 Summary

This chapter presents a framework SOIM- Semantic Inference based Short Text Messages Awareness [55] , which further enhances the search capabilities of SAMS, as presented in the Chapter-3, via semantic reasoning inference capabilities enrichment.

In this chapter we presented an enhancement to the work that has been done in the Chapter-3. The SOIM framework is developed on top of SAMS, which facilitates short text message annotation and semantic reasoning capabilities with contextual domain ontology. It is built on semantic web technologies, in which a generic ontology of a set of concepts and properties that define the most popular query keywords is developed in order to improve the framework search and reasoning process, of which extra knowledge about an input message query could be inferred to help the awareness of message context. The similarity degrees are calculated in terms of the generic ontology in order to match the relevant patterns with an input query.

The implementation of the proposed framework was presented to validate its feasibility in terms of efficiency and accuracy in matching and retrieval. Number of experimental tests has been conducted to evaluate the SOIM performance against SAMS. The results have shown a significant enhancement with respect to the SOIM. However, choosing the best retrieved pattern based on a similarity degree only might increase the degree of misclassification error of choosing the right matched pattern. Thus, Fuzzy lattice reasoning is suggested to tackle this issue which will be discussed in detail in the next chapter.

CHAPTER 5

5 FUZZY LATTICE REASONING BASED SOIM

5.1 Overview

Fuzzy Logic theory is one of the appropriate techniques that are adopted to deal with ambiguous, imprecise, noisy, or missing contextual information. It addresses the degree of uncertainty for context inference and facilitating the decision-making [41] [64] . In this chapter the effectiveness of Fuzzy Lattice Reasoning has been employed on top of the SOIM framework. The framework SOIM was presented in the Chapter 4 to tackle the mobile text message awareness problem and provide semantic based query inference capabilities. Its reasoning is based on calculating the semantic distance between input message keywords in terms of the domain ontology to find the degree of match of each keyword with its corresponding abstract class in domain ontology and then compute the similarity degree of aggregating message keywords with respect to the patterns.

The issue of choosing the best-retrieved pattern that matches input message keywords arises because of the degree of similarity of the inferred patterns would be nearly close to each other regarding their values. Consequently, this affects the decision making of the system to classify the input message query to its right pattern category. For this purpose, rules based a fuzzy reasoning classifier that adopt the Fuzzy Set theory for decision making is chosen to deal with uncertainty, which arises when the boundaries of a class of objects are not sharply defined.

The performance of the proposed classifier is evaluated through various experimental tests and compared with some other classifiers with respect to the classification accuracy and time. In addition, precision and recall along with the probabilistic evaluations are utilised during the system evaluation.

5.2 Methodology

Obviously, the importance of the properties on deducing fine-grained actions is higher than the importance of the ones that infer less specific actions. Here, we introduce a model for incorporating fuzzy logic with specific fuzzy rules and inherent semantics in the SOIM framework elaborated in the last chapter. It is applied to improve the decision making capability of the framework and reduce the misclassification error.

To increase the accuracy of the classification process with clearer decisions, the classifier generates specific fuzzy rules to strengthen the decision on determining the object class for an input short text message ***Q*.** For instance, consider that ***Q*** matches the patterns A_k with a degree of similarity which is equal to 0.7, A_l with a similarity degree of 0.65, and as A_m with 0.38. based on the rule of the winner takes all (i.e the pattern with higher value), the decision goes to the action with the higher degree of similarity disregarding the possibility of contradicting, the abstraction of properties, and incompatible matching. Hence, to achieve a better qualitative performance a fuzzy reasoning classifier based rules that adopt the Fuzzy Set theory for decision making is defined (see Figure 5-1).

The FLR attains the advantages of hyperbox-based rule induction including fast computation and straightforward interpretation. In addition, it cope with hyperbox disadvantages such as the restriction in unit hypercube $[0,1]^N$ via the applicability in both complete and non-complete lattices including the Euclidean space R^N , a lack of introducing tuneable non-linearities through adopting sigmoid membership function and lastly, the decision-making outside a hyperbox using the potential for tuneable generalization beyond a hyperbox. Further advantages the FLR classifier providing of a capability for incremental learning, granular computing and dealing with missing data.

During training, classifier FLR deals with a series of training data pairs (a,c), where *c* is the class label of data *a*. it is interpreted as rule 'if antecedent (*a*) then consequent (*c*)'. The FLR then adapts a hyperbox-shaped region of fuzzy training sets according to a principle of minimal elongation of the hyperbox diagonal to classify the datum according to their classes' labels. While in testing phase the classifier employs the previous rules induced during training phase to categorize any new unknown antecedent.

5.2.1 Fuzzy Data Modelling

Enhancing the inference capability of learning and classifying context of a fuzzy classifier is determined by the deduced rules of classification and the training dataset representation. In the present study, the data representation that would be fed to the classifier can act as:

Consider *n numerical* contextual dataset (input parameters) $p_i \in P$, $i = 1, . . . , m$ and patterns labelled as $y_i \in Y$, $i = 1, . . . ,n$. To specify rules for classification, a trained fuzzy system express each training pair (*p*,*y*) as a fuzzy lattice rule ⟨u,c⟩, where u is an element of the fuzzy lattice ⟨U,μ⟩ and c is the corresponding class. Thus, every instance in the training set could be treated as fuzzy lattice rules.

As aforementioned in the previous chapter, the contextual information in SOIM are hierarchically indexed as a set of labelled modules in terms of domain ontology structure, *i.e.,* the sub concept on level *l* represents more specific context than the concepts in lower levels it derived from, for example meeting at level 1, which refers to ground concept is less specific meaning than business meeting at level 2 and so on. The hierarchical indexing *(i.e.,* the level index of the ingredients) is used as a part of input parameters *p* to the FLR classifier along with the degree of matches and the degree of similarity. Basically, the combination of these parameters orients the classifier direction towards the right pattern. For example, lets y_l and y_k, are two inferred patterns that almost have the same similarity degree with input *p*, but they differ in their semantic distances in terms of the ontology. The classifier here can adjust its rules based on the obtained similarity degree in addition to the parameters levels, and therefore, the pattern that has the same level number or the closest one to input *p* would be the best match to be the winner by the classifier.

The input data for the classifier was obtained from the outputs of SOIM framework, where the results of the degree of match among the input query P and the predefined set of saved pattern along with the their similarity degree is regarded as the input parameters to FLR classifier. However, during the inferring time, the reasoner might infer all the predefined patterns that share the input query with the same module even though with deferent levels. Thus, the degree of similarity of the inferred patterns would be nearly close to each other in their values. Consequently, this affects on the decision making of the system to follow the right pattern to do the right action accordingly.

5.2.2 Fuzzy Classification Training

To efficiently create a system identification and classification that fit input–output contextual situation (Message query ⇒(Pattern, Action)) with less misclassification error, a fuzzy trained system is suggested with a set of fuzzy classification rules, of which certain patterns are labelled to specific events or context labels. In learning phase the classifier trains set of predefined patterns $\{u_1, u_2, \ldots, u_M\} \subseteq U$, each one associated with a set of labelled classes $C=\{c_1, c_2, \ldots, c_K\}$, $c \in C$, hence a set of fuzzy lattice rules are induced through implementing the function f:U→C. The sigmoid function is utilised as a positive valuation function to map an interval of real numbers to a fuzzy lattice. It is a non-linear increasing function that applied to normalized and non-normalized data, hence, it is adopted in fuzzy lattice [47]

In this work, the estimation of right pattern by the classifier is based on a given sigmoid positive valuation function, as in the Equation (5.1), with isomorphic function, as in the Equation (5.2), for non-normalized lattice R, and the inclusion measure that is defined by the function in the Equation (5.1) [71] .

$$v_i(x)= 1/(1+\exp(-\lambda(x-x_m)) \qquad (5.1)$$

Where, $x_m= (x_{min}+x_{max})/2$, $\lambda=\varsigma/(x_{max}-x_{min})$, $\varsigma>0$, and x_{min} , x_{max} are the minimum and maximum attribute values in the corresponding constituent lattice of the training data [71] [47] [70] .

$$\theta_i(x)= 2x_m-x \qquad (5.2)$$

The capacity of the sigmoid non-linearity function in R^N has improved the classification performance because of the tuneable generalisation. This means most the testing data lies inside rule hyperbox core region. Hence, compared to the unit-hypercube, the classifier FLR that applicable in R^N is better with respect to the efficient performance.

For an unknown input text message Q of n numerical contextual input properties, $Q = [q_1, \dots , q_n]$, a fuzzy rule that concludes the right pattern correspond to Q follows the condition: if q_i satisfies the rule $\langle u_{i1},c_i\rangle$,……, *and* q_n satisfies the rule $\langle u_{in},c_n\rangle$,then Q is classified as a class C. The rule above holds for all the antecedents of input patterns query and the predefined patterns. The algorithm in the Figure 5-2 below shows the classification trail to identify new input query.

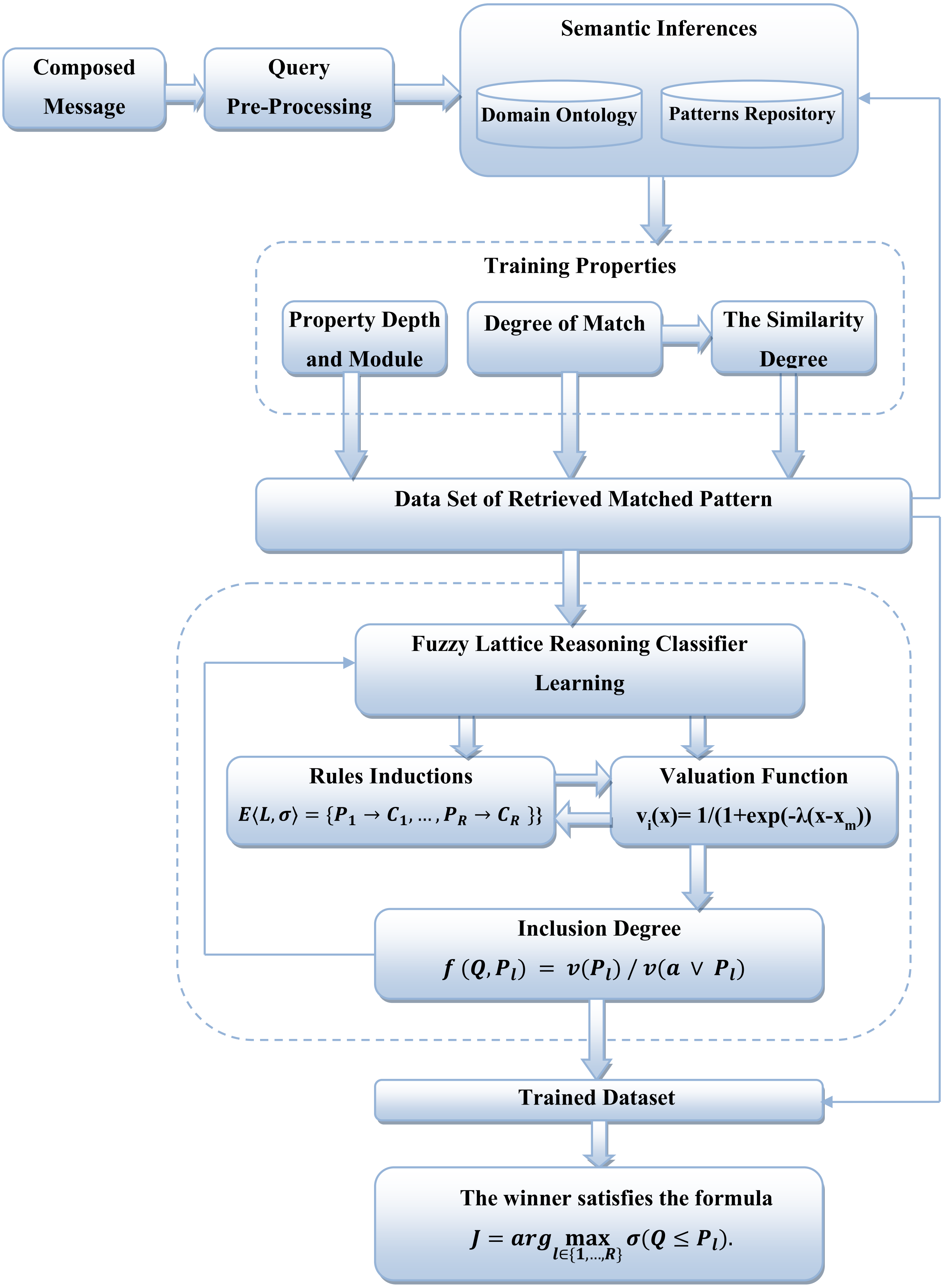

Figure 5-1: Fuzzy Lattice Reasoning (FLR) Structure

Input: Input query vector Q=[q_1, …., q_n]

Set of predefined trained patterns with high degree of similarity, C_k, k= $1,\ldots,m$

A concluded fuzzy lattice rule engine Let $E\langle L,\sigma\rangle = \{P_1 \to C_1, \ldots, P_R \to C_R\}$.

Output: The winner pattern with best match $Q \to C_l, l \in k$.

Step-1: Present Q to the rules engine set $E\langle L,\sigma\rangle$.

Step-2: Based on the valuation function (5), calculate the fuzzy degree of inclusion $\sigma(Q \le P_l), l \in 1,\ldots,R$.

Step-3: Repeat Applying step 2 to the all antecedents P_l that defined in $E\langle L,\sigma\rangle$.

Step-4: Compare the obtained degree of inclusion of the all rules in $E\langle L,\sigma\rangle$, the winner is the rule that satisfy the formula, $J = arg \max_{l\in\{1,\ldots,R\}} \sigma(Q \le P_l)$.

Step-5: The output will be Q→C_l, that is, the antecedent Q is clustered in $P_l \to C_l$.

Figure 5-2: FLR algorithm to choose the winner pattern

Table 5-1: The attributes of dataset

	Attributes Name	Symbol	Data Type
1	1st Property Depth	P_1	Real number
2	2nd Property Depth	P_2	Real number
3	3rd Property Depth	P_3	Real number
4	4th Property Depth	P_4	Real number
5	1st Best Match Degree	D_1	Real number
6	2nd Best Match Degree	D_2	Real number
7	3rd Best Match Degree	D_3	Real number
8	4th Best Match Degree	D_4	Real number
9	Belief Measure	S_d	Real number
	Class Label	M_c	*'High', 'acceptable', 'Low' Similarity*

Table 5-2: The classification accuracy results of FLR classifier compared to other classifiers

Classifier Name	Classification Accuracy (%)		Number of Rules	
	With Missing Data	Without Missing Data	With Missing	Without Missing
FLR –Linear	80.45	82.35	7	5
Fuzzy - ART	N/A	84.1	N/A	8
C4.5	79.34	77.45	6	12
FLR- Sigmoid	80.75	83.33	5	3

Table 5-3: FLR classification results with missing data

Parameter value (ς)	Parameter value (ρ)	Classification Accuracy (%)		Number of Rules
		Training Set	Testing Set	
1	0.3	61.52	71.45	8
	0.5	68.67	79.22	10
	0.7	62.11	72.24	8
5	0.3	60.36	79.70	5
	0.5	63.24	80.75	5
	0.7	65.36	78.72	4

Table 5-4: FLR classification results without missing data

Parameter value (ς)	Parameter value (ρ)	Classification Accuracy (%)		Number of Rules
		Training Set	Testing Set	
1	0.3	59.50	66.00	3
	0.5	61.80	66.00	3
	0.7	63.34	83.33	3
5	0.3	58.20	73.52	3
	0.5	62.80	77.22	2
	0.7	62.00	68.45	5

5.2.3 Experimental Results and discussion

A set of experiments to test the framework evaluation in terms of classification accuracy and the processing times for training and testing was compared with three other classifiers, namely, the fuzzy-ART classifier, C45 and the FLR classifier with a linear positive valuation function. The classifiers selection was made based on the similarity aspects with the FLR classifier with respect to the rules- based classification, hypercube regions and data with missing values. A set of 500 patterns are used for training and 150 patterns are used for testing which varies in the values of attributes to learn the classifier with respect to the all dataset attributes.

The chosen classifiers have been tested on same selected dataset and compared with the FLR classifier. The conducted experiments demonstrate the capacities of the FLR classifier against the other classifiers in terms of a given parameter values range. Table 5-2 shows the accuracy of classification of the tested classifiers with regards to the datasets with and without missing values. The results are calculated based on the values of the slop and vigilance parameters with ς (1, 5) and ρ (0.3, 0.5, 0.7) respectively, which can be used for tuning the slope of evaluation function. According to obtained results, the fuzzy-ART classifier, which works on hyperbox handling, performed 84.1% with 8 of induced rules. Similarly, the performance of the C4.5 classifier was 77.45% and 79.34% for data with and without missing values while, the numbers of induced rules were 12 and 6 rules for data without and with missing values respectively. A positive valuation function (5.1) and isomorphic function (5.2) in each constituent lattice R is carried out by the FLR classifier. Compared with the aforementioned classifiers, the FLR classifier has performed better in the capacity for generalization on non-normalized data (see the Table 5-2). An accuracy of 83.33% with parameter values (ς=5, ρ=0.7) has been resulted without missing data values, while for data with missing values an accuracy of 80.75% with (ς=5, ρ=0.7) was obtained. Similarly, fewer rules are induced considerably (three rules for data without missing and five rules with-missing values as shown in the Table 5-3 and 5-4.

In spite of the limited trained datasets, the accuracy percentages of the FLR classifier have shown better results than the corresponding percentages of the same classifier with linear function and C4.5 classifier. The results have shown that the accuracy rate of Fuzzy-ART classifier was slightly higher than the FLR classifier due to the limited amount of training/testing dataset. However the performance of FLR was much better and faster than the

C4.5 classifier and the Fuzzy-ART in this application but with less deduced rules as depicted in the Table 5-2. Additional advantages of the FLR classifier is its capability of employing tuneable sigmoid non-linearities for improving performance and faster training process using a single pass [47] . Therefore, in this application, the FLR classifier performance is better than the other classifiers in the R^N non-linearities space.

Table 5-2: The FLR classification rules

Rule No.	1st Prop	2nd Prop	3rd Prop	4th Prop	1st DOM	2nd DoM	3rd DoM	4th DoM	Sd	Decision
1	[0.0 , 124.0]	[0.0, 124.0]	[0.0, 124.0]	[0.0, 124.0]	[0.6, 1.0]	[0.6 , 1.0]	[0.0 , 1.0]	[0.0, 1.0]	[0.68, 1.0]	High Similarity
2	[0.0 , 124.0]	[0.0, 124.0]	[0.0, 124.0]	[0.0 , 124.0]	[0.4 1.0]	[0.4, 1.0]	[0.8, 1.0]	[0.7, 1.0]	[0.68, 1.0]	Acceptable Sim
3	[0.0, 124.0]	[0.0, 124.0]	[0.0, 124.0]	[0.0, 124.0]	[0.0, 0.5]	[0.0 0.7]	[0.0 , 1.0]	[0.1, 1.0]	[0.0,0.5]	Low Similarity

All the experiments have been carried out on a Pentium 4 processor at running at 1.68 GHz frequency and 1GB of RAM. The Waikato Environment for Knowledge Analysis (WEKA) has been used to carry out the experiments with C4.5 using the J48 algorithm. Matlab is used to conduct the Fuzzy-ART classifier. The FLR classifier has been implemented using WEKA [111] APIs on Java environment.

5.3 FLR based SOIM Evaluation

The fetched dataset contains retrieved patterns that were retrieved by SOIM in the Chapter 4, each pattern has four properties, the highest five match degree values with respect to the input query properties and its similarity degree. Each property value refers to its depth with respect to the given domain ontology hierarchy. Therefore, the hierarchy of message properties (their depth in domain ontology), the similarity degree and the match Degree of Measure (DoM) are contributed in the evaluation method of F-SOIM as stated in the Table 5-1, in which the attributes of the dataset are declared. The target variable in the dataset record is the best pattern with high similarity to the received unknown query. To tackle this problem and infer the most matched patterns, the FLR classifier is experimented using two datasets:

the first one is used without missing values, while the second one has some missing values. in the experiments of the present study, a sigmoid positive valuation function was utilised since it contributed in increase the classification accuracy compared with linear function. The framework is compared with SAMS matching module and the SOIM- semantic based matching module, respectively in terms of accuracy in patterns retrieval and discovery. Initially, the effective of fuzzy set to leverage the accuracy of choosing similarity degrees of the relevant patterns to a message query is considered. The domain ontology representation is used for evaluating the similarity measures. Standard metrics of Precision P and Recall R, are employed. Based on belief measure that expresses how the input query q are similar to pattern p with respect to the similarity metric (as discussed in the Section 4.5.1), the retrieval effectiveness is examined to show how well a similarity reasoning process deduces relevant patterns as they have been specified by the user.

According to the test that has been conducted in the Chapter 4, we applied SOIM and F-SOIM with the same set of patterns. Two tests were performed to evaluate the efficiency of the framework enhancement; the first test was performed on patterns that have no missing values whereas, the second test examines the set of patterns with one missing attribute. Each test was repeated for ten times in random with respect to the patterns order. Each test contains a set of 50 patterns, 4 properties and 20 patterns were relevant to a text message query. The Figure 5-3 and 5-4 shows that F-SOIM has outperformed the SOIM. This is because enriching the F-SOIM with hierarchy semantics and specialization have affected in the similarity metric efficiency. From the results we conclude that using Fuzzy affects on the performance of SOIM positively in spite of the enhancement is marginally better in the case of missing attributes as illustrated in the Figure 5-4.

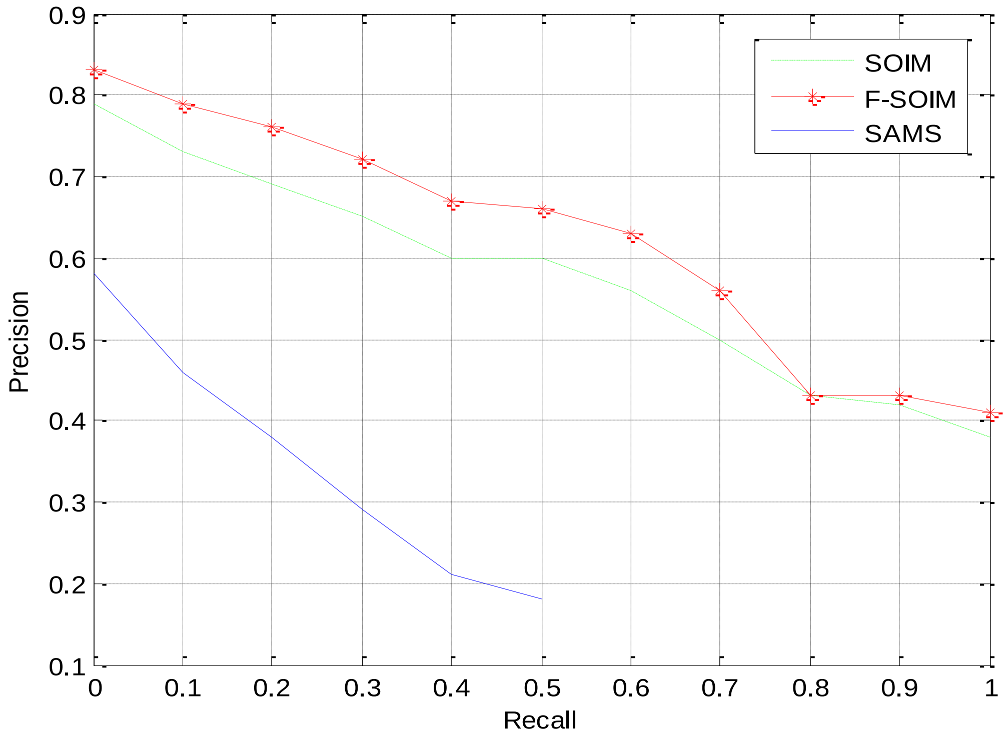

Figure 5-3: The performance of F-SOIM against SOIM (test1)

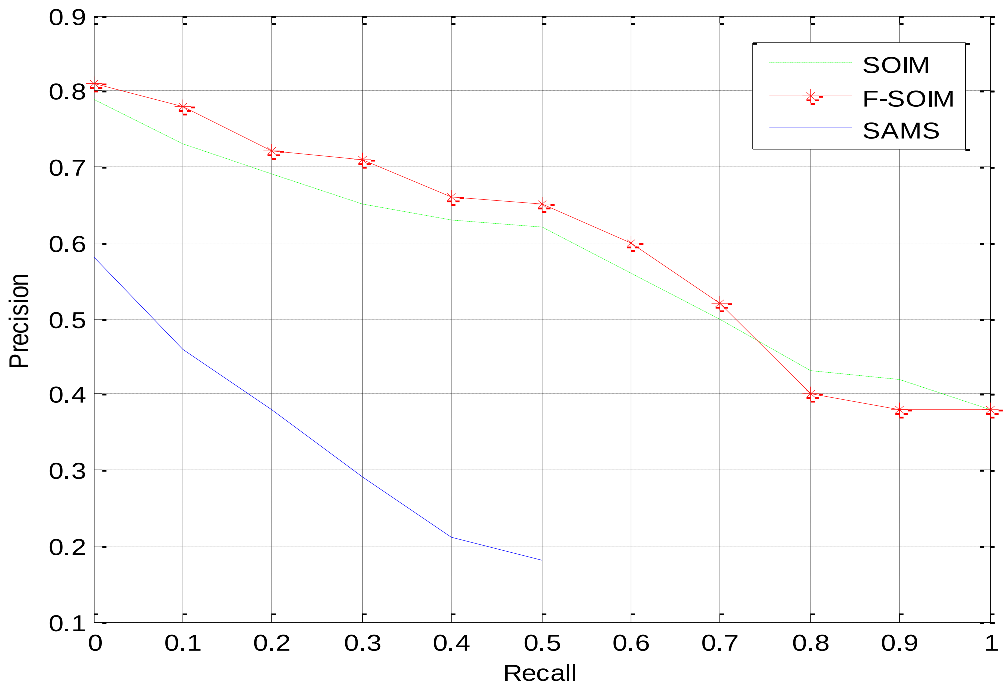

Figure 5-4: The performance of F-SOIM against SOIM (test 2)

5.4 Probabilistic Evaluation

Based on the outcomes of the search, in which two possible values can have, p (probability of success) and q (probability of success), the efficiency of a system can be assessed by computing the probability of successes. The Binomial Distribution $b\ (x;\ n,\ p)$, as defined by the Equation (5.3), is calculated to give a good approximation for SOIM and F-SOIM, based on number of trials.

$$b(x, n, p) = \binom{n}{x} p^x\ q^{n-x} \qquad (5.3)$$

Let x_{so}, n_{so} , p_{so} and b_{so} are the number of successes, number of trials, the success probability and the Binomial distributed variable, respectively, for the SOIM framework. Correspondingly, x_{fs}, n_{fs}, p_{fs} , and b_{fs} are the parameters and the Binomial distributed variable that used for F-SOIM.

Table 5-3: The Binomial Distribution for SOIM and F-SOIM

No of Successes	10	20	30	40	50	60	70	80	90	100
b_{fs}	0.000	0.000	0.0000	0.0000	0.0000	0.0002	0.0340	0.0629	0.0002	0.000
b_{so}	0.000	0.000	0.0000	0.0000	0.0000	0.0000	0.0027	0.0962	0.0062	0.000

A comparison is carried out by computing the probability of getting x successes for a number of trials for the both frameworks, in which, 100 patterns trials were carried out for each of the two frameworks. Based on the first success probability of p_{fs} is 0.81 and p_{so} is 0.76 (obtained from Geometric distribution calculation – see Section 3.8.2 in the Chapter-3), the number of successes increases up to 80, so, the value of b_{fs} is increased against the value of b_{so}. This means that the success probability p is higher for F-SOIM compared to SOIM as shown in the results in the Table 5-4 and the Figure 5-4.

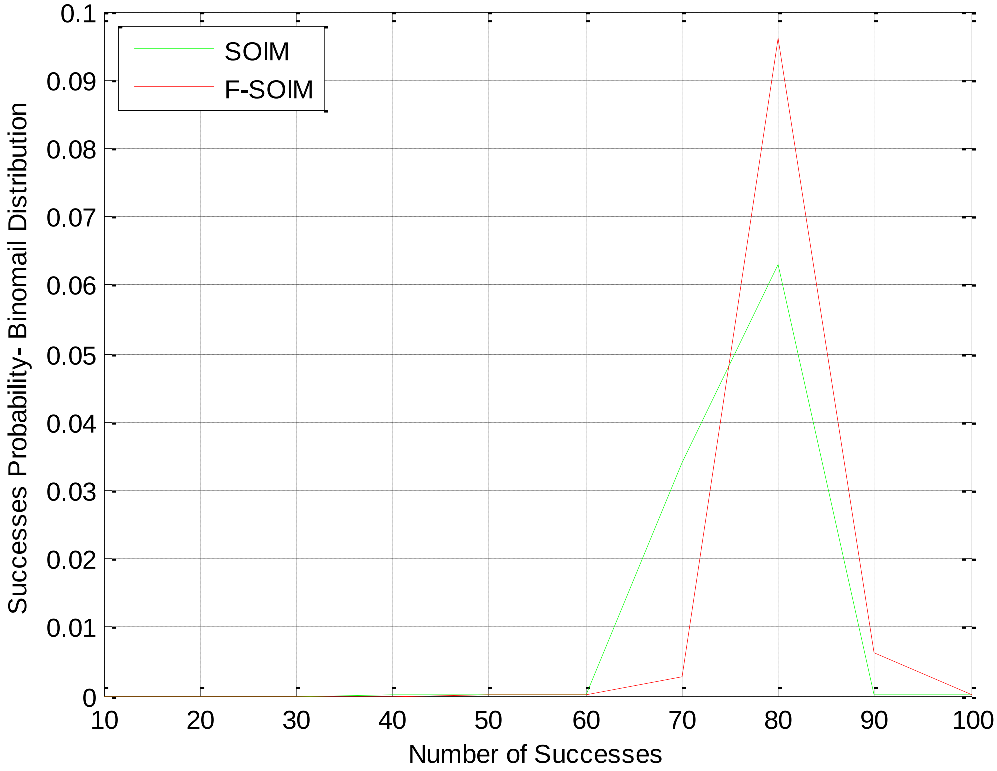

Figure 5-5: Successes probability using Binomial Distribution

5.5 Summary

This chapter presented a fuzzy lattice reasoning classifier that is applied on top of the SOIM for enhancing query deduction. It tackled the issue of choosing the best-retrieved pattern that matches input message keywords by adopting the hierarchy of message query ingredients with respect to ontology. The FLR classification has shown better performance than the other chosen classifiers in terms of accuracy, classification time and induced rules' numbers. In spite of the limited trained datasets, the results were encouraging for further research evaluation to be carried out through dealing with varying large number of new unknown messages dataset to assess the message awareness.

CHAPTER 6

6 INDECISIVE PROPERTIES REDUCTION USING ROUGH SETS THEORY

6.1 Overview

Discriminating the importance of properties is one of the significant issues in data analysis research as it concentrates on discovering dependencies between attributes. Dependent properties that are used in a set of predefined patterns may not affect the match results with a new text message query; hence, they are dispensable in matching process. One challenging task is addressing uncertainty issue of patterns' properties when matching with other received message queries. Thus, in this chapter we investigate the redundancy and dependency of patterns properties using the Rough Sets theory. Applying Rough Sets on top of the SOIM framework is evaluated with set of experiments and compared the results against the framework's previous evaluations in the last chapters in order to present the effectiveness of removing redundancy and dependency on the proposed framework's performance.

6.2 Redundancy and Dependency

One of the significant issue in data analysis is discovering dependencies between attributes since it helps to discriminate the importance of properties [145] . Dependent properties are indecisive properties that are dispensable in matchmaking[76] . Dependency between two sets can be represented as $P \Rightarrow Q$, in which a set of attributes Q depends on a set of attributes P totally this is achieved when attribute values from P include values of attributes from Q in a degree of k $(0 \leq k \leq 1)$ based on the Equation (6.1) [78] [77] :

$$k_P(Q) = \frac{|POS_P(Q)|}{|U|} \quad (6.1)$$

A reduct γ is the minimal attributes set in which the irrelevant attributes are reduced but accomplish the same quality of classification as with the original attributes set before reduction [74] [78] . It is also defined as the relation between a subset S of the conditional attributes set C, in which:

$$\gamma S(D) \ = \ \gamma C(D) \quad (6.2)$$

A given dataset may have many mini attributes reduct sets, so the set R of all reducts is defined as [73] [77] :

$$R = \{X : X \subseteq C, \gamma S(D) = \gamma C(D)\} \tag{6.3}$$

Dependent properties can be expressed more precisely in terms of reduct as follows:

Let A, B be a set of properties, where $B \subseteq A$. And a be a property belong to B $a \in B$, then [77] :

- *if* $IND(B) = IND(B - \{a\})$ *then* a is *dispensable* in B
- B is *independent* if all its attributes are indispensable.

Any subset C of B is regarded as a *reduct* if the above conditions are met (i.e. C is independent and $IND(C) = IND(B)$). the *core of attributes* as defined in (6.4) refers to the intersection of all the sets in R, where it is the set of all indispensable attributes [78] [74] .

$$Core(B) = \cap R(B) \tag{6.4}$$

6.2.1 Query Modeling with rough set

The framework considers a set of predefined patterns properties and new input message query properties to apply Rough Sets redundancy and dependency techniques before performing matching module. Figure 6-1 shows the components for using Rough Sets as a part of the framework to matchmaking**.** Both the text message query and the selected predefined patterns are fetched to Irrelevant Property Reduction module to remove irrelevant properties of the predefined patterns in terms of ontology. Dispensable properties are discovered and reduced by the Dependent Property Reduction module.

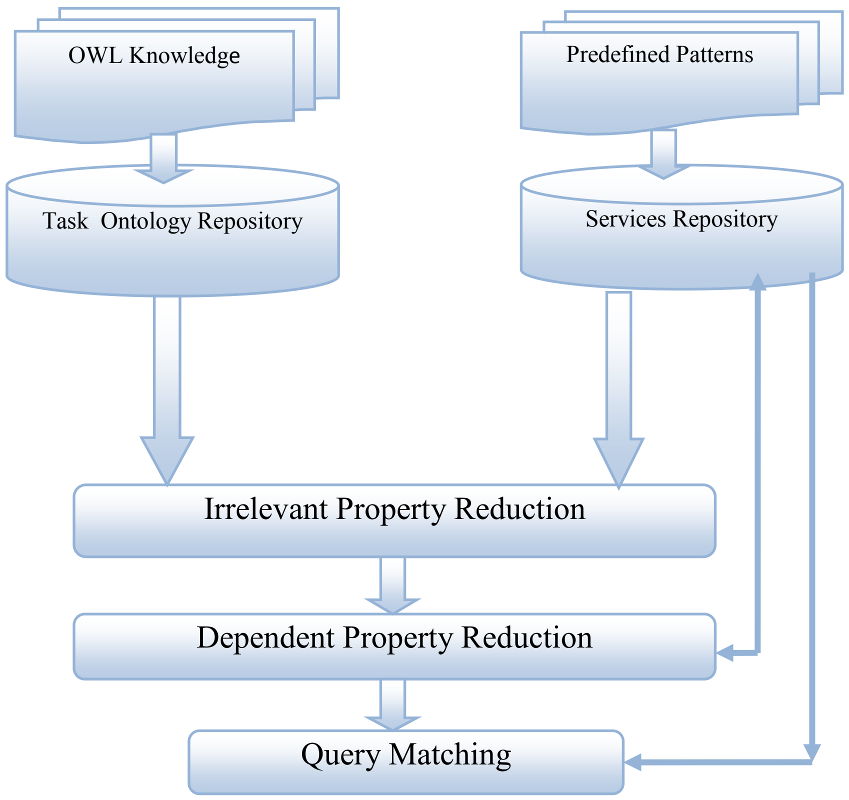

Figure 6-1: Rough Sets module

To assist the system in choosing the proper set of patterns that maximally satisfy the message query, the lower and upper approximations of Rough Sets are applied to the set.

Let:

- U *be* a set of N patterns (the universe of all the predefined patterns), $U = \{s_1\ s_2, \ldots, s_N\}, N \geq 1$.
- P be a set of K properties used to describe the N patterns of the set U, $P = \{ p_1, p_2, \ldots, p_K\}, K \geq 2$.
- P_A be a set of M chosen patterns properties relevant to a input query Q in terms of the task ontology O (defined in chapter 4), $P_A = \{ p_{A2}, p_{A1}, \ldots, p_{AM} \}, P_A \subseteq P$, $M \geq 1$.
  - X be a set of patterns relevant to the query Q (that match query to some extent), $X \subseteq U$.
  - $\underline{X}$ be the lower approximation of the set X.
  - $\overline{X}$ be the upper approximation of the set X.

From the aforementioned definition of the approximations, the lower and upper approximations are declared as:

$$X = \{x \in U\colon [x]_{PA} \subseteq X\} \tag{6.5}$$

$$X = \{x \in U\colon [x]_{PA}\ X \neq \emptyset\} \tag{6.6}$$

For a property $p \in P_A$, we get the following:

- $\forall x \in \underline{\boldsymbol{X}}$, x definitely has property p .
- $\forall x \in \overline{\boldsymbol{X}}$, x possibly has property p .
- $\forall x \in U - \overline{\boldsymbol{X}}$, x absolutely does not have property p .

The above approximations are used to provide query with an approximate match by encoding properties, which cannot be represented exactly.

6.2.2 Reducing Irrelevant Properties

The notion of reduction is to remove indecisive attributes from the set of properties of which the same quality of classification as the original set is gained. . Since the query is a human natural language request so that it may has a number of irrelevant properties, *e.g.*, ***<your body health check will be on Monday 22nd June -2011 at 10 am in hillingdon hospital>***. In this example we can find some stop words like (*will, in,on*..) as well as some irrelevant words like (*body*). Initially, the pre-processing module in the framework removes all stop words at the time of annotation. Whereas, the properties (keywords) with match degree of zero with respect to the predefined patterns, are removed before the matchmaking process is carried out, hence, the properties of the query after removing stop words and irrelevant words will be in the form of ***<health, check, Monday, 22nd June 2011,11 am, hillingdon, hospital>***. Following the work done in the Chapter 4, let:

- p_Q be a property for a input query.
- p_A be a property for a selected pattern.

Input: Query properties pQ ,pQ =[$q1$, …., qn]

Set of predefined patterns, input for each property pA, pA= $1,….,m$ yin one of patterns.

Output: All pA that are marked with no match removed from pattern.

Step-1: Do Loop1- for all selected predefined patterns p_A.

Step-2: Do Loop2- for all properties used in p_Q .

Step-3: Apply equation (4-) to match p_A with p_Q, in terms of the domain ontology O.

Step-4: compare the match value of p_A and p_Q based on the defined relations (*exact, plugIn, subsumed, uncertain,* and *nomatch*).

Step-5: End Loop2

Step-6: **If** p_A is no match with any p_Q **Then** marked it with *nomatch* .

Step-7: End Loop1 – Repeat steps (2-5) until no more patterns to be compared.

Step-8: All p_A with *nomatch*

Figure 6-2: Reducing irrelevant properties from predefined patterns

As shown in the Figure 6-2, for every property of a query, the reduction module treats with those properties with which no match results as indecisive properties and removes them from the saved patterns that are organised as a set of records in a patterns repository (see Chapter-4). The process of reduction boosts the matching process. All irrelevant properties are organized in a record in such away each column refers to specific property a given pattern. Some properties columns might have empty values because a property used by one selected pattern might not be used by another one. Accordingly, the record column with an empty value becomes an uncertain property with respect to a query, whereas, the record column associated with the property that had not match any other property in a query, it is marked as *nomatch* in the record. Consequently, all properties with empty values *i.e.,* uncertain properties are not considered in matchmaking module.

6.2.3 Dependant Attributes Reduction

Data analysis explores dependencies discovering between attributeswhere, he saved patterns may have dependent properties. As shown in the Figure 6-3, the Dependent Property Reduction module performance navigates predefined patterns to explore any decisive properties. Thus, indecisive properties which may include uncertain properties too are discovered during this stage. Accordingly, it further obtains the targeted patterns, which are the ones, can be identified without using the indecisive properties. Furthermore, the performance of the whole proposed framework can be speeded up because of the reduction process at the time of properties matching.

Let [76]

- $P_A{}^D$ be a set of *decisive* properties, in which,

$$P_A{}^D = \{ P_{A1}{}^D, P_{A2}{}^D, \dots , P_{An}{}^D \}, P_A{}^D \subseteq P_A , n \geq 1.$$

- $P_A{}^I$ be a set of *indecisive* properties, in which,

$$P_A{}^I = \{ P_{A1}{}^I, P_{A2}{}^I, \dots, P_{Am}{}^I \}, P_A{}^I \subseteq P_A , m \geq 1.$$

- $P_A{}^{I\_Core}$ be a core subset of $P_A{}^I$ that has the maximum number of individual indecisive properties and the group of these properties in $P_A{}^{I\_Core}$ are indecisive in identifying advertised services relevant to the message query Q in terms of O , $P_A{}^{I\_Core} \subseteq P_A{}^I$.

- $R\,(P_A{}^I)$ be an equivalence relation also called indiscernibility relation on U .

- f be a decision function discerning an advertised service with properties.

Then

$$\mathrm{R}\,(P_A{}^I) = \{(\mathrm{x},\mathrm{y}) \in \mathrm{U}: \forall\, P_{Ai}{}^I \in P_A{}^I , f(\mathrm{x}, P_A{}^I) = f(\mathrm{y}, P_A{}^I)\} \quad (6.7)$$

$$P_A{}^D = P_A - P_A{}^I \quad (6.8)$$

Input: M is a set of patterns that are relevant to input query.

P_A is a set of properties used by the set M

$P_A{}^D$ is a set of decisive properties, $P_A{}^D \subseteq P_A$

$P_A{}^I$ is a set of individual indecisive properties, $P_A{}^I \subseteq P_A$

$P_A{}^{I\_Core}$ is a set of combined indecisive properties, $P_A{}^{I\_Core} \subseteq P_A{}^I$

Initialization: $P_A{}^D = \emptyset$

$P_A{}^I = \emptyset$

$P_A{}^{I\_Core} = \emptyset$

Processing:

Do Loop

If p, $p \in P_A$, is an indecisive property in the set M **Then**

$p \Rightarrow P_A{}^I$

$P_A{}^{I\_Core} = \emptyset$

$p \Rightarrow P_A{}^{I\_Core}$

End If

End Loop

Do Loop $(2 \le I \le \text{sizeof}(P_A{}^I)-1)$

Calculate all possible *i* combinations of the properties in $P_A{}^I$

If *i* is indecisive property in M **Then**

$P_A{}^{I\_Core} = \emptyset$

$i \Rightarrow P_A{}^{I\_Core}$

Else If *i* is decisive property **Then** Break

End If

End Loop

$P_A{}^D = P_A - P_A{}^{I\_Core}$

Output: return $P_A{}^D$

Figure 6-3: Reducing dependent properties algorithm

6.3 Discussion and Evaluation

A set of experiments is conducted to evaluate R-SOIM and the results are discussed in the following sub-sections. In this section, we evaluated its performance in terms of the accuracy and efficiency in retrieval matched patterns and compared its results against the work which has been presented in the last chapters.

6.3.1 Experiment Results

In term of the domain ontology shown in the Figure 6-4, a set of patterns was chosen randomly from the patterns repository (see Chapter-4) to construct a pattern decision table for a specific new query, of which dependent properties among patterns can be identified. Choosing specified pattern is based on a condition, in which, at least one of its properties has a relationship (*exact, plug-in, or subsumed)* with a property used in an incoming message query. According to the following properties of query :

Query: **<*InternelM, PhoneN, Date, Meetingroom, FormalM, vAddress, Building, Email*>.**

Table 6-1: Selected patterns for decision table

Properties / Patterns	c10	d8	c7	b11	b5	c4	d9	b3	d5	b8	d12
R1	1	X	1	X	X	1	1	1	1	1	1
R2	X	X	1	X	X	1	X	1	X	X	1
R3	1	1	X	1	X	1	X	1	X	1	X
R4	1	1	X	1	1	1	1	X	X	1	X
R5	X	1	X	X	X	X	X	1	1	X	X
R6	1	X	1	X	1	1	X	1	X	1	X
R7	X	X	X	1	X	1	X	1	X	X	X
R8	1	1	X	X	X	1	1	1	X	1	X
R9	X	X	1	1	X	1	1	1	X	X	1
R10	1	X	1	X	X	1	X	X	1	X	1
R11	X	1	1	1	X	1	X	1	1	1	X
R12	1	X	1	1	X	1	X	1	1	X	1
R13	X	1	X	X	1	X	1	X	X	1	X
R14	1	1	1	1	X	1	1	1	1	X	1
R15	1	X	1	1	X	X	X	1	1	X	X

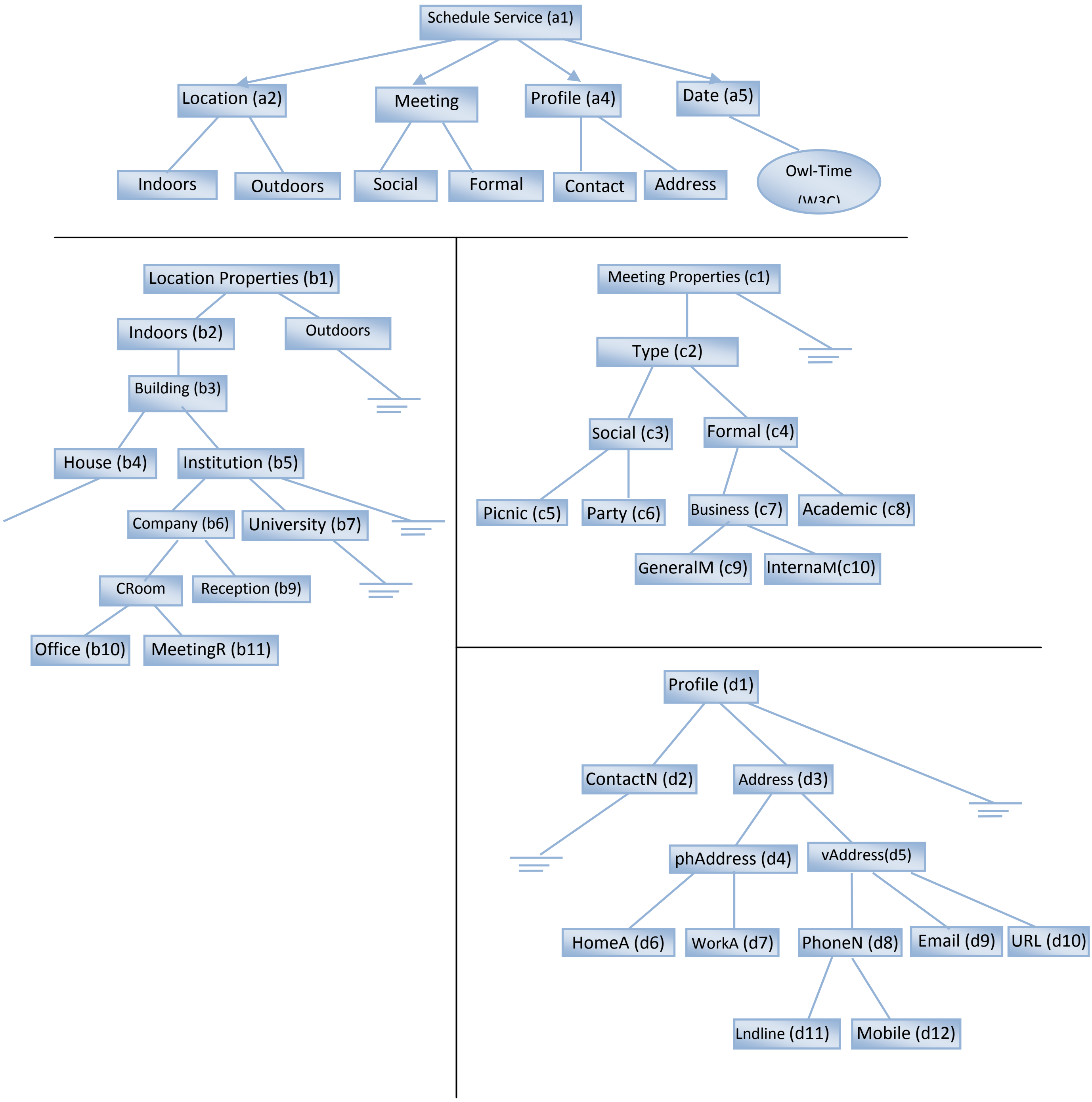

Figure 6-4: Domain ontology used in patterns retrieval

15 relevant patterns records were chosen for decision as shown in the Table 6-1, namely c10, d8, c7, b11, d9, b5, c4, b3, d5, b8 and d12. The content of table can be discussed as:

- A record tagged with 1: means the property in that record is explicitly used by the pattern, *e.g.*, the properties c10, c7, d9, c4, b3, b5, b8 and d12 are declared in the pattern R1.

- A record marked with X: it means the corresponding property is not involved in the given pattern. However, a property marked with X might be dependent on other properties used by the pattern, i.e. it does not necessarily to be irrelevant to the pattern but could be dependent.

By using the dependent properties reduction algorithm shown in the Figure 6-2, we find the properties d9, c4 and b3 are recognized as dependent properties (indecisive) that are eliminated when calculating the degree of belief and removed from the decision table (see Table 6-1 with shaded dependent properties). The table of patterns with decisive properties (see Table 6-2) is used for computing the match degree (using the Equation 4.1, presented in the Chapter-4) between each decisive property used in pattern entry and a property in the new input message with respect to the domain ontology that declared in the Figure 6-4. It is worth noting that a match degree of 50 percent is marked with X in the Table 6-1 to refer to the uncertain properties. Also, the similarity degree of the query against the selected patterns with decisive properties is computed using the Equation 6.3 in the Chapter-4. The Figure 6-5 illustrates the results of similarity degree of R-SOIM in comparison with the SAMS, SOIM and F-SOIM.

Table 6-2: The match degree of decisive properties in the decision table

Degree of Match	100%	100%	55%	100%	78%	100%	87%	34%
Properties	c10	d8	c7	b11	b5	d5	b8	d12

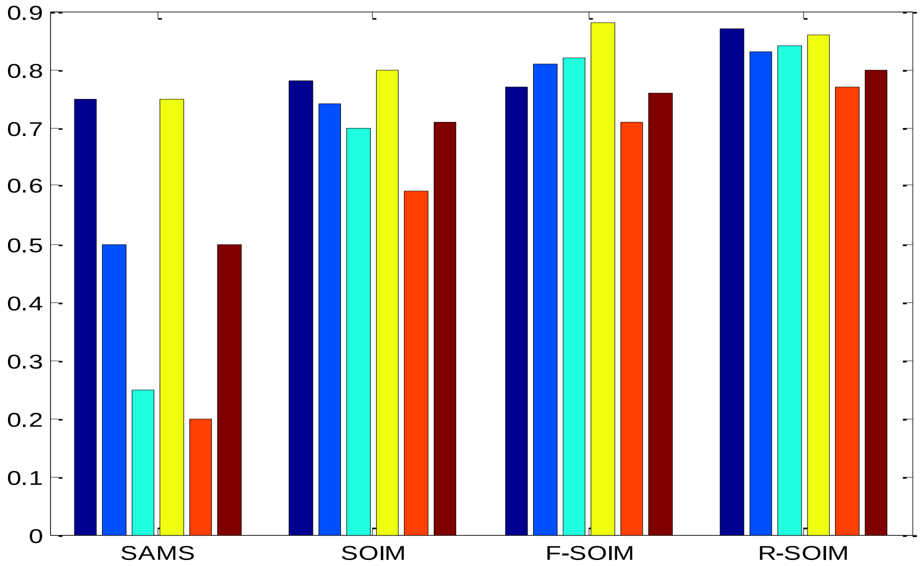

Figure 6-5: The similarity degree of R-SOIM compared to previous the framework's stages

6.3.2 Precision and Recall

To assess the performance of R-SOIM, we used the same dataset that has been implemented in the Chapter-4. We prepared two group of 10 tests have been prepared for test the framework. In each test list of 50 patterns were produced in a random order. Each list had at least eight patterns which were relevant to an input query. In addition, five properties are included in each pattern with at least two of them were dependent properties. Two constraints were enforced on the selected patterns in group one in order to ensure that the relevant patterns are returned by SAMS, SOIM, F-SOIM and R-SOIM matching respectively. In the first constrain, one property of a pattern at least has an exact match, while the second with no pattern and has an uncertain property. The selected patterns in group two were released from the two constrains, *i.e.,* some of the patterns did not have properties of exact match and some other patterns have uncertain properties. Moreover, the patterns ranked with degree of similarity of no significant value (zero) are not retrieved because the irrelevant patterns' algorithm removes all the insignificant properties.

As observed in the Figure 6-6, which shows the averaged results of the tests of group one, we notice that R-SOIM shows the best performance against others *i.e,* SAMS, SOIM and F-SOIM because of the dependency and redundancy capabilities along with the semantic reasoning inferences provision. In the tests of group two, F-SOIM has shown better performance in few cases due to its capability of not dealing with uncertain properties and thus some irrelevant patterns that had uncertain properties were not returned as the Figure 6-7 shown. However, R-SOIM has shown good performance in most other cases. In addition, since SAMS and SOIM have different limitations in patterns matching, therefore, their recall do not accomplish 100 percent in the tests of group two , *e.g.,* SAMS matched 7 of the 10 relevant patterns, SOIM and F-SOIM matched 8 relevant patterns. We also evaluated the performance of R-SOIM with respect to both the groups as plotted in the Figure 6-8. The Figure shows that R-SOIM performance is varied between group one and two because of the use of uncertain properties, which may affect on the relevant patterns retrieving process.

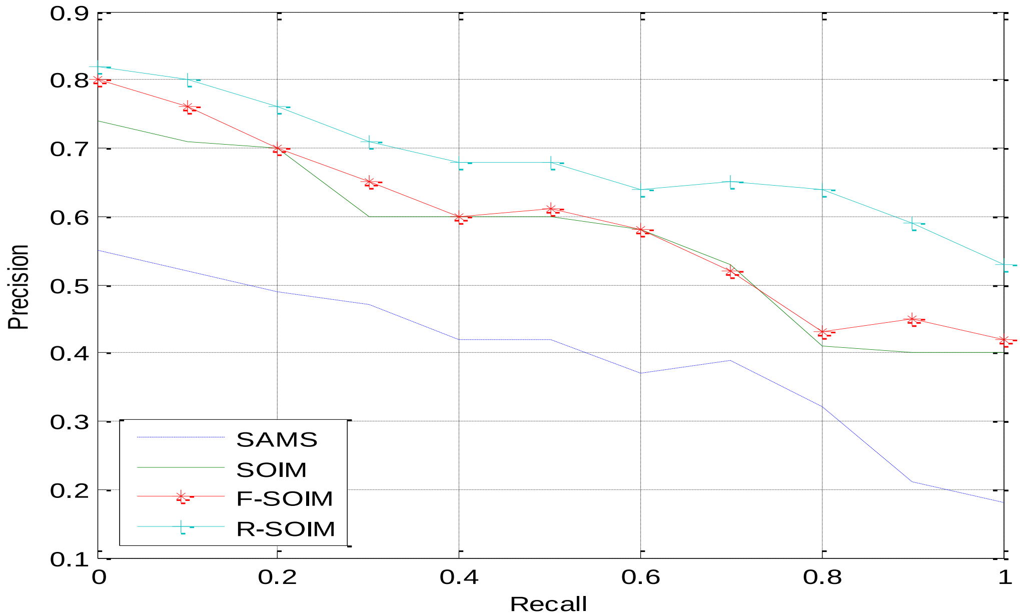

Figure 6-6: The performance of R-SOIM, F-SOIM, SOIM and SAMS – group 1

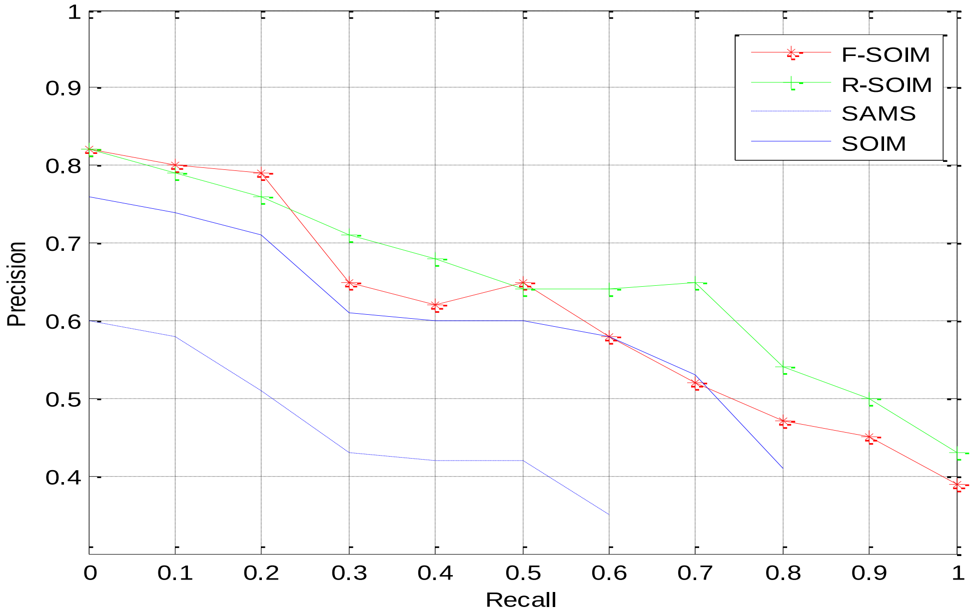

Figure 6-7: The performance of R-SOIM, F-SOIM, SOIM and SAMS – group 2

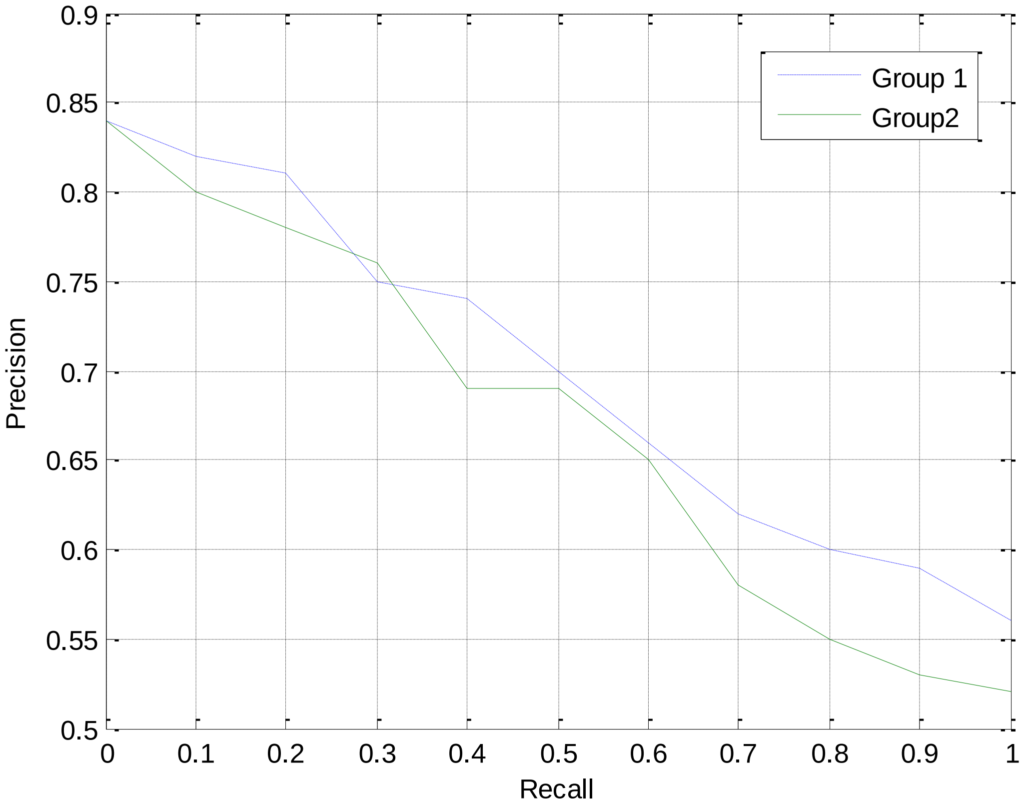

Figure 6-8: The performance of R-SOIM (group 1 and group 2)

Table 6-3: Binomial distribution of R-SOIM with compared to SOIM, F-SOIM

No of Successes	10	20	30	40	50	60	70	80	90	100
b_{rs}	0.000	0.000	0.0000	0.0000	0.0000	0.0001	0.0237	0.0764	0.0004	0.000
b_{fs}	0.000	0.000	0.0000	0.0000	0.0001	0.0197	0.0791	0.0027	0.0000	0.000
b_{so}	0.000	0.000	0.0000	0.0000	0.0000	0.0012	0.0870	0.0108	0.0000	0.000

6.3.3 Probabilistic Evaluation

The efficiency of R-SOIM is assessed by computing the Binomial Distribution $b(x; n, p)$, to estimate its probability of successes with compared to SOIM and F-SOIM. Based on the test that has been done in the Chapter 5, we repeated it with the same number of trails and the probability for x success is calculated for SOIM, F-SOIM and R-SOIM when the value of p_{rs} is 0.82, p_{fs} is 0.81 and p_{so} is 0.76. The result of 98, 97 and 91 patterns were returned successfully of the R-SOIM, F-SOIM and SOIM respectively. the probability of success for the R-SOIM is higher with respect to less number of trials in comparison with the other frameworks as The outcomes of binominal distribution is shown in the Table 6-3 and the Figure 6-9.

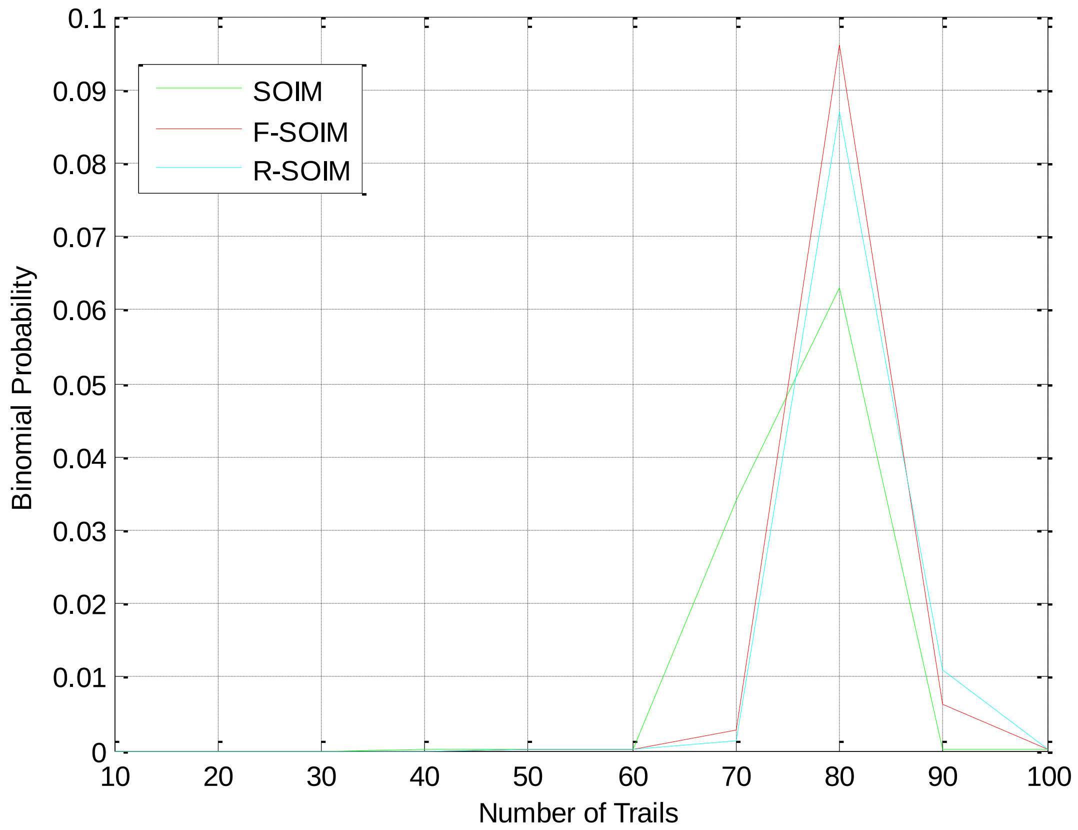

Figure 6-9: Successes probability using Binominal Distribution

6.4 Summary

This chapter has presented Rough Sets theory in reduction process. Rough Sets is built on top of the SOIM framework for uncertain properties reduction of a set of selected patterns that are relevant to a received message query depending on the approximations- lower and upper. This means that with little certain properties about patterns can verify better performance in a system discovery and accurate retrieval.

The measures precision and recall were used to examine the performance of the framework SOIM with Rough Sets. The performance of R-SOIM showed better results in terms of retrieval accuracy compared with the same framework without using the Rough Sets concepts. Other tests were carried out to evaluate the effectiveness of using rough sets in SOIM such as Binomial probability distribution.

CHAPTER 7

7 CONCLUSIONS AND FUTURE WORK

7.1 Overview

In this chapter, the whole thesis is summarized and conclusions are drawn. Further, suggestions for the future work are also presented. The thesis gives an in-depth possibility of enhancing the performance of the SOIM framework to manage the text messages semantically. Main motivation for investigating the research of Message-awareness arises from the notion of rendering a handheld device intelligent enough to interact with users seamlessly and materialize the ambient intelligence environment. For further research and suggestions, the future work section highlights those research areas where the findings of this research could be further investigated and improved.

7.2 Conclusion

The work done in this thesis has investigated automatic annotation of short text message reasoning with context ontology on low-end mobile devices. We presented a prototype to leverage short text messages with a set of internal and external metadata, which facilitate inference process for classification purposes, without needing learning process.

The proposed research adopts domain ontology with a set of ontological modules. These ontological modules are linked to the message's keywords and metadata, where a text message is automatically annotated with existing ontology. This system does not need learning process because it is not relying on the training set to infer a text message. This task was achieved by structuring the domain ontology modules as a set of categories that can be expressed as combinations of contexts, in which each module refer to a specific context, situation or event category. Predefined patterns, which represent a given particular domain ontology modules, are defined. This accomplishes the task of tagging the inferred text to a particular pattern in terms of ontology matching, which might help in the next potential matching process for new unknown messages without the needs for ontology navigation.

Practical frameworks are developed for facilitating short text message annotation and ontology based semantic inferences. The proposed SAMS framework automatically annotates the composed message during its creation time with the corresponding attributes. Further, it is assembled in an XML file and is aggregated with metadata in order to make it feasible for publishing in a networked environment with a set of authorised devices. The SAMS copy on each connected device can parse the received file and search the incoming query with a set of predefined patterns to match and retrieve a proper action. The lightweight XML parser, namely kXML, is used to parse the XML file. Several MIDlets in the JME platform were implemented by developing two case studies to validate the framework feasibility. Experimental evaluations were tested on SAMS to assess its performance. The proposed SAMS framework structure is based on crisp matching process; hence, it was necessary to add further improvements in it. A semantic based framework, SOIM, which facilitates mobile messages with annotation, discovery and publishing using semantic reasoning of a message query, was proposed. The SOIM framework is built on semantic web technologies by using ontology, which is mainly used for reasoning and expressing contextual information. It adds up extra knowledge in the deduction process. The SOIM framework `s reasoning is based on calculating the degree of similarity between input query and the predefined set of patterns in terms of ontology. Precision, recall, and probabilistic evaluations were presented to evaluate SOIM and to determine its accuracy and efficiency. The framework SOIM reasoning is based on calculating the semantic distance between the input message keywords in terms of the domain ontology to calculate the keyword similarity with its corresponding abstract class in domain ontology. The degree of similarity of the inferred patterns would be nearly close to each other regarding their values and hence, the issue of decision making of the best-retrieved pattern that matches input message keywords arises. In order to classify the input message query to its right pattern category, rules based a fuzzy reasoning classifier were adopted for decision making. Discovering dependencies between attributes is a significant issue in data analysis as it helps to discriminate the importance of properties. To deal with properties dependency and uncertainty, which may affect the system performance, Rough Set theory was applied to reduce the uncertainty of pattern properties, when matching set of selected patterns, with message query by eliminate any irrelevant properties which are indecisive properties and are dispensable in matchmaking. The usability of Rough Sets theory has approved in data processing as it does not need any preliminary or additional information about data compared with the approaches that are based on Dempster Shafer theory and Bayesian networks[131] [132] .

The overall results of the proposed framework are encouraging as they were experimentally evaluated in terms of the measures Precision, Recall and Probabilistic Distribution. The results have shown that the framework gives better improvements when FLR and Rough Set theory are applied to the system, which proves the applicability of an context ontology for semantic short text messages reasoning. However, the proposed categorization method is not without weaknesses. Limitations of the proposed domain ontology for specific situation has affected the system generalisation, but we believe that as rich and comprehensive domain ontologies become more available in future, the proposed approach may prove to be suitable for different kind of categorizations. In addition, the proposed method relies on the factual knowledge from the ontology by concentrating on the message content and therefore may miss other important contextual interpretations that do not match the user's perceived interests. In this context, the system can be enhanced by learning from user reaction to the reasoning decision and validating message classification accordingly.

7.3 Future work

Intelligent devices research is still an under explored area. This research was specifically focused on the field of intelligent messaging service on hand held devices. However, many significant aspects remain open for further research and need to be explored to further leverage the research work. Thus, some of these aspects are recommended in this section for research direction in Semantic messaging awareness Following are some of the future research directions within the work carried out in this thesis, which may offer opportunities for further investigation within ongoing research.

7.3.1 The Designed Algorithm

A number of suggestions for improvement to the designed algorithm were achieved. However, the development of algorithms that are based on the knowledge and skills of the researcher still depends on the availability of resources and implementation platforms, which ultimately prolonged the work. Therefore, further enhancement in the proposed designed algorithm may be considered for future studies. In addition, the proposed algorithm can be extended to develop the concept of virtual secretary (E-Secretary).

7.3.2 System Security

Since, the investigation of the security issues is beyond the scope of our presented research in this thesis, therefore, there exists a potential to improve the user reliability and trust. Enhancing the proposed algorithms to incorporate the security aspects requires exploration of a set of points including the device check and sender's authority to share the resources.

7.3.3 Light-weighted Reasoner

Hand-held devices, such as mobile phones, have limited computational power and other resources which impose the constraints on the performance of a reasoning task in realistic time. To carry out our research, the reasoning task in our presented work was performed on a network server. Facilitating semantic applications on the resource limited hand-held devices and developing a distributed light weight efficient reasoned can be an interesting future study.

7.3.4 The Research Environment

OWL language and JME platform were adopted to develop the SOIM framework. For improvement of reasoning and retrieval performance in a connected environment, the interoperability between different ontology languages can also be explored in the future research. Moreover, the possibility of employing other mobile platforms other than JME for comparison purposes is applicable.

7.3.5 The Ontology Alignment

In this thesis, the possibility of ontology alignment has not been explored. To further, increase the accuracy of retrieval and discovery. The ontologies that are defined for the same domain can be grouped together using the alignment process. The existing efforts on ontology alignment could enhance the research and the framework may take the advantage of the previous alignments, specifically from the alignments in similar domains. The search mechanism in the SOIM framework can be extended to take advantage of multiple ontologies as it employs single ontology for extracting additional information about query keywords in this research.

REFERENCES

[1] Pan G., Xu Y., Wu Z., Yang L., Lin M. and Li S. (2011) 'Task Follow-me: Towards Seamless Task Migration Across Smart Environments', *IEEE Intelligent Systems*, 26(3) pp. 50-57.

[2] Chen H., Finin T. and Joshi A. (2004) 'An ontology for context-aware pervasive computing environments', *The Knowledge Engineering Review*, 18(3) pp. 197-207.

[3] Bodyanskiy Y. and Shubkina O. (2011) 'Semantic Annotation of Text Documents Using Modified Probabilistic Neural Network', *Intelligent Data Acquisition and Advanced Computing Systems, IEEE 6th International Conference,* Prague, 15-17 September pp. 328-331.

[4] Fan X. and Wei D. (2011) 'A Method of Agent and Patient Relation Acquisition for Short-Text Classification', *Communication in Computer and Information Science*. 153(1) pp. 27-33.

[5] Ferragina P. andScaiella U.(2010) 'TAGME: On-the-fly Annotation of Short Text Fragments (by Wikipedia Entities)', *Proceedings of the 19th ACM international on Information and knowledge management,* Toronto, Canada, 25–29 October, pp.1625-1628.

[6] Rafeeque P C, Sendilkumar S. (2011) 'A Survey on Short Text Analysis in Web', *Advanced Computing (ICoAC), 2011 Third International Conference,* Chennai,14-16 December, pp. 365-371.

[7] Vitale D., Ferragina P. and Scaiella U. (2012) 'Classification of Short Texts by Deploying Topical Annotations', *Advances in Information Retrieval,* , Lecture Notes in Computer Science, vol. 7224 pp. 376-387.

[8] Wang X., Gao H., Hu X. and Liu H. (2012) 'Enriching short text representation in microblog for clustering', *Frontiers of Computer Science*, 6(1) pp. 88-101.

[9] Wei H., Shanfei L., Yuejin T. and Bing G. (2009) 'Association Rules Based Short Text Feature Extension', *International Journal of Computer Science and Network Security IJCSNS*, 9(10) pp. 227-230.

[10] Zavitsanos E., Tsatsaronis G. and Varlamis I. (2010) 'Scalable semantic annotation of text using lexical and web resources', *Artificial Intelligence: Theories, Models and Applications,* Lecture Notes in Computer Science, vol. 6040 pp. 287–296.

[11] Knox S., Shannon R. , Coyle L., Clear A. K., Dobson S., Quigley A. J. and Nixon P. (2008) 'Scatterbox: Context-Aware Message Management', *Revue d'intelligence artificielle*, pp. 1-20. Available at: http://www.csi.ucd.ie/UserFiles/publications/1210610073042.pdf

[12] Ho J. and Intille S. S. (2005) ' Using context-aware computing to reduce the perceived burden of interruptions from mobile devices', *CHI 2005: Proceedings of the SIGCHI conference on Human factors in computing systems*: *ACM Press*, Portland, Oregon, USA, -7 April, pp. 909-918.

[13] Zelikovitz, S. and Hirsh, H. (2000) 'Improving Short Text Classification Using Unlabeled Background Knowledge', *Seventeenth International Conference on Machine Learning (ICML)*, Stanford. Available at: ftp://athos.rutgers.edu/http/http/pub/zelikovi/icml00.pdf

[14] Bloehdorn, S. and Hotho, A. (2004) 'Text Classification by Boosting Weak Learners based on Terms and Concepts', *4th IEEE International Conference on Data Mining (ICDM'04)*, pp. 331 – 334.

[15] Berners-Lee T. and Cailliau, R. (1990) 'WorldWideWeb: Proposal for a HyperText Project', [Online] Available at: http://www.w3.org/Proposal.html (Accessed: 10 December 2011).

[16] Smith M. K., Systems E., D., Welty C. and Mcguinness D. L. (2004) 'OWL Web Ontology Language Guide', [Online] Available at: http://www.w3.org/TR/owl-guide/ (Accessed: 20 June 2009).

[17] Steller L. and Krishnaswamy S. (2008) 'Optimised Mobile Reasoning for Pervasive Service Discovery', *IEEE International Conference on Web Services,* Beijing, 23-26 September 2008, pp. 789-790.

[18] Hong J., Suh E. and Kim S. (2009) 'Context-aware systems: A literature review and classification', *Expert Systems with Applications*, 36(4) pp. 8509-8522.

[19] Li M., Yu B., Rana O., Wang Z. and Member S. (2008) 'Grid Service Discovery with Rough Sets', *Knowledge Creation Diffusion Utilization,* 20(6) pp. 851-862.

[20] Cui Y. (2007) 'Messaging design and beyond: learning from a user study on holiday greeting messages', *Proceedings of the 4th international conference on mobile technology, applications, and systems and the 1st international symposium on Computer human interaction in mobile technology*, NewYork, USA, pp. 159–166.

[21] Porter M.F. 'An algorithm for suffix stripping, Computer Laboratory, Cambridge', [Online] Available at: www.emeraldinsight.com/0033-0337.htm (Accessed: 15 October 2009).

[22] Cimiano P., Ladwig G. and Staab S. (2005) 'Gimme: the context: context-driven auto- matic semantic annotation with c-pankow', In *Proceedings of the 14th World Wide Web Conference* (IW3C2), Chiba, Japan, 10-14 May 2005, pp.332–341.

[23] Ding Y. and Embley D.W. (2006) 'Using data-extraction ontologies to foster automating semantic annotation', *Data Engineering Workshops, 2006. Proceedings. 22nd International Conference*, Atlanta, GA, USA, 24 April 2006, pp. x138.

[24] El-Beltagy S.R., Hazman M. and Rafea A.A. (2007) 'Ontology based annotation of text segments', *SAC'07 Proceedings of the 2007 ACM symposium on Applied Computing*, New York, NY, USA, pp. 1362-1367.

[25] Laclavik M., Seleng M., Gatial E., Balogh Z. and Hluch (2007) 'Ontology based Text Annotation --OnTeA', Proceedings of the 2007 conference on Information Modelling and Knowledge Bases XVIII, Amsterdam, The Netherlands, pp. 311-315.

[26] Erdmann M., Maedche A., Schnurr H.P. and Staab S. (2001) 'From manual to semi-automatic semantic annotation: About ontology-based text annotation tools', Published on 30 December 2001 by Linko¨ping University, Electronic Press, Linko¨ping, Sweden.

[27] Corcho O. (2006) 'Ontology based document annotation: trends and open research problems', *International Journal of Metadata, Semantics and Ontologies*, 1(1) pp. 47-57.

[28] Misund G. and Lindh M. (2006) 'Annotating mobile multimedia messages with spatiotemporal information', *Journal of Geographic Information Sciences*, 19(2).

[29] C. Thurlow (2003) 'Thurlow & Brown 'Generation Txt? The sociolinguistics of young people's text-messaging', [Online] Available at: http://extra.shu.ac.uk/daol/articles/v1/n1/a3/thurlow2002003.html (Accessed: 13 June 2010).

[30] Figliola P. M. and G. Stevens (2011) 'Text and Multimedia Messaging: Emerging Issues for Congress', [Online] Available at: http://www.ipmall.info/hosted_resources/crs/RL34632_110622.pdf (Accessed: October 2011).

[31] Britain D. (2009) 'Department for Culture, Media and Sport and Department for Business , Enterprise and Regulatory Reform Digital Britain The Interim Report', [Online] available at: http://www.official-documents.gov.uk/document/cm75/7548/7548.pdf (Accessed: September 2008).

[32] Neches R., Fikes R., Finin T., Gruber T., Patil R.,Senator T. S. and Swartout W. R. (1991) 'Enabling Technology for Knowledge Sharing,' *AI Magazine*, 12(3), [Online] Available at: http://www.isi.edu/isd/KRSharing/vision/ **AIMag.html.**

[33] Hu D. H., Dong F. and Wang C. L. (2009) 'A Semantic Context Management Framework on Mobile Device', *International Conference on Embedded Software and Systems*, Zhejiang , pp. 331-338.

[34] Cabral L., Domingue J. and Motta E. 'Approaches to Semantic Web Services: An Overview and Comparisons', University of Southampton, [Online] Available at: http://eprints.soton.ac.uk/262688 (Accessed: June 2009).

[35] Wagner M., Paolucci M., Luther M., Boehm S., Hamard J. and Souville B. (2006) 'Contextual Intelligence for Mobile Services through Semantic Web Technology', *The 3nd European Semantic Web Conference (ESWC 2006)* Budva, Montenegor, 12th May 2006.

[36] Coyle K. ' Understanding Metadata and its Purpose'*Journal of Academic Librarianship*, 31(2), pp 160-163. [Online] Available at: http://kcoyle.net/jal-31-2.html.

[37] Cathro W. (1997) 'Metadata: An Overview', *In Standards Australia Seminar*: Matching Discovery and Recovery', [Online] Available at: **https://www.nla.gov.au/openpublish/index.php/nlasp/article/viewArticle/1019/1289** (Accessed October 2011).

[38] Sen A. (2004) 'Metadata management: past, present and future', *Decision Support Systems*, 37 (1) pp. 151–173.

[39] Costello R. L. and Jacobs D. B. (2003) 'A Quick Introduction to OWL Web Ontology Language What is OWL', [Online] Available at: http://www.iwayan.info/Lecture/PerangkLunakAplkWEB_S2/Onto&OWL/CostelloQuickIntroOwl.pdf.

[40] Noy N. F. and McGuinness D. L.. (2001) 'Ontology development 101: A guide to creating your first ontology', *Technical Report*,Stanford University, Stanford, CA, 94305
. [Online] Available at: http://protege.stanford.edu/publications/ontology_development/ontology101.pdf.

[41] Anagnostopoulos C. and Hadjiefthymiades S. (2008) 'Enhancing Situation-Aware Systems through Imprecise Reasoning', *IEEE Transactions on Mobile Computing*, 7(10) pp. 1153-1168.

[42] Dey A., Hamid R., Beckman C., Li I. and Usu D. (2004) 'A CAPpella: programming by demonstration of context-aware applications', *Proceeding CHI04,* 24-29 April 2004, Vienna, Austria, pp. 33–40.

[43] Padovitz A., Loke S., Zaslavsky A., Burg B. and Bartolini C. (2005) 'An approach to data fusion for context-awareness', *Modelling and Using Context*, Lecture Notes in Computer Science, vol. 3554, pp 9-63.

[44] Matheus M., Kokar K., Baclawski J. and Letkowski C. (2004) 'The Practical Application of Semantic Web Technologies for Situation Awareness', *Proceeding of IEEE Eighth Int'l Symp. Wearable Computers (ISWC)*.

[45] Himberg J., Mantyjarvi J. And Korpipaa P. (2001) 'Using PCA and ICA for exploratory data analysis in situation awareness', *Proceeding of IEEE MFI 2001*, pp. 127–131.

[46] Anagnostopoulos C. and Hadjiefthymiades S. (2010) 'Advanced fuzzy inference engines in situation aware computing', *Fuzzy Sets and Systems*, 161(4) pp. 498-521.

[47] Kaburlasos V. G., Athanasiadis I. N. and Mitkas P. A. (2007) 'Fuzzy lattice reasoning (FLR) classifier and its application for ambient ozone estimation', *International Journal of Approximate Reasoning*, 45(1) 152-188.

[48] An Oracle White Paper (2010) 'Introduction to Java Platform: Enterprise Edition', [Online] Available at: http://www.oracle.com/us/products/middleware/application-server/050871.pdf (Accessed: June 2011).

[49] Sun Microsystems Inc. (2003) 'Connected Limited Device Configuration (CLDC)-', Specification, version 1.1, Java™ 2 Platform , Micro Edition, (J2ME™), [Online] Available at: http://java.sun.com/products/cldc/ (Accessed: October 2011).

[50] Sun Microsystems Inc., 'Connected Limited Device Configuration (CLDC) Specification', version 1.1, Java™ 2 Platform , Micro Edition, (J2ME™),

2000. Available at: http://java.sun.com/products/cldc/ (Accessed: January 2010).

[51] Java Community Process (2002) 'Mobile Information Device Profile for Java™ 2 Micro Edition', Version 2.0, JSR 118 Expert Group. [Online] Available at: http://www.oracle.com/technetwork/java/index-jsp-138820.html (Accessed: October 2009).

[52] Knudsen O. (2002) 'Parsing XML in J2ME', [Online] Available at http://developers.sun.com/mobility/midp/articles/parsingxml/ (Accessed: October 2009).

[53] Ghosh S. (2003) 'Add XML parsing to your J2ME applications' , [Online] Available at: http://www.ibm.com/developerworks/library/wi-parsexml/ (Accessed: November 2009).

[54] Ricci F. (2010) 'Java 2 Micro Edition XML', http://www.inf.unibz.it/~ricci/MS/slides-2010-2011/7.1-J2ME-XML.pdf (Accessed November 2009).

[55] Java Community Process 'JSRs: Java Specification Requests – JSR 75: PDA Optional Packages for the J2ME Platform', Available at: http://jcp.org/en/jsr/detail?id=75. (Accessed: March 2011).

[56] Al-sultany G., Li M., Jan S. and Al-raweshidy H. (2011) 'Intelligent Mobile Messaging', *Fuzzy Systems and Knowledge Discovery (FSKD), 2011 Eighth International Conference*, 26-28 July 2011, Shanghai, vol. 4 pp.2671- 2674.

[57] Reynolds D. 'Jena 2 Inference support', [Online] Available at: http://www.iis.uni-stuttgart.de/lehre/ss04/xml/Jena-2.1/doc/inference/ (Accessed March 2009).

[58] Carroll J. J., Dickinson I., Dollin C., Reynolds D., Seaborne A., and Wilkinson K. (2004) 'Jena: Implementing the Semantic Web Recommendations', *Proceeding WWW Alt. '04 Proceeding of the 13th International World Wide Web Conference on Alternate Track Papers & Posters*, NewYork, NY, USA, pp.74-83.

[59] Davies J., Fensel D., Harmelen F. V. (2003) 'Towards the Semantic Web-Ontology-driven Knowledge Management', John Wiley & Sons Ltd, San Francisco, USA.

[60] Schneider L. (2003) 'How to Build a Foundational Ontology', *KI2003: Advances in Artificial Intelligence*, Lecture Notes in Computer Science, vol. 2821 pp. 120- 134.

[61] Xiao M., Hu J. and Xiao Y. (2007) 'A Study on Ontology Learning for the Intelligent Search Engine', *International Conference on Wireless Communications, Networking and Mobile Computing*, September 2007, pp. 5364-5367.

[62] Li M., Yu B., Rana O. and Wang Z. (2008) 'Grid Service Discovery with Rough Sets', *IEEE Transactions on Knowledge and Data Engineering*, 20(6) pp. 851-862.

[63] Su L. T. (1994) 'The relevance of recall and precision in user evaluation', *Journal of the American Society for Information Science*, April 1994, 45(3) pp. 207-217.

[64] Jia B., Zhong S., Zheng T. and Liu Z. (2010) 'The Study and Design of Adaptive Learning System based on fuzzy set theory', *ACM digital System*, pp. 1-11.

[65] Quan T. T., Hui S. C. and Fong A. C. M. (2006) 'Automatic Fuzzy Ontology Generation for Semantic Help-Desk Support', *IEEE Transactions on Industrial Informatics*, 2(3) pp. 155-164.

[66] Choi S. and Yang H. (2004) 'A Fuzzy Set Based Tutoring System for Adaptive Learning', *Adaptive Hypermedia and Adaptive Web- Based Systems*, Lecture Notes in Computer Science, vol. 3137 pp. 389-392.

[67] KaburlasosV. G., Athanasiadis I. N. and Mitkas P. A. (2007) 'Fuzzy Lattice Reasoning Classifier and its Application for ambient ozone estimation', *International Journal of Approximate Reasoning*, 45(1) pp. 152-188.

[68] Guo B. and Ska I. V. (2006) 'Mathematical framework for lattice problems', *International Journal of numerical analysis and modelling,* 1(1) pp.1-36.

[69] Silverman J. H. (2006) 'An Introduction to the Theory of Lattices and Applications to Cryptography', *Summer School on Computational Number Theory and Applications to Cryptography*,University of Wyoming.

[70] Athanasiadis I. N., Kaburlasos V. G., P. Mitkas A. and Petridis V. (2003) 'Applying Machine Learning Techniques on Air Quality Data for Real-Time Decision Support*', First International Symposium on Information Technologies in Environmental Engineering (ITEE 2003),* June 2003, Gdansk, Poland, pp. 51.

[71] Athanasiadis I. (2006) 'Air quality assessment using fuzzy lattice reasoning (FLR)', *Fuzzy Systems, 2006 IEEE International Conference*, Vancouver, BC, 11 September 2006, pp. 29-34.

[72] Witten I. H. and Frank E. (2000) 'WEKA – Machine Learning Algorithm in Java', Chapter 8, Morgan Kaufmam Publisher, pp. 268.

[73] Komorowski J., Polkowski L. and Skowron A. 'Rough Sets: A Tutorial', [Online] Available at: http://secs.ceas.uc.edu/~mazlack/dbm.w2011/Komorowski.RoughSets.tutor.pdf (Accessed: January 2011).

[74] Jensen R. and Shen Q. (2003) 'Finding rough set reducts with ant colony optimization', *Proceedings of the 2003 UK workshop on Computational Intelligence.*

[75] Duntsch I., Gediga G. (1999) 'Rough set data analysis',, School of Information and Software Engineering, University of Ulster.

[76] Yu B. and Li M. (2006) 'RSSM: A Rough Sets based Service Matchmaking Algorithm', Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.99.7384&rep=rep1&type=pdf.

[77] Pawlak Z. (1991) 'Rough Sets', *Institute of Theoretical and Applied Informatics*, Polish Academy of Sciences, University of Information Technology and Management, pp. 1-51.

[78] Rissino S. and Lambert G. (2009) 'Rough Set Theory – Fundamental Concepts Principals Data Extraction and Applications', *Data Mining and*

Knowledge Discovery in Real Life Applications, Book edited by: Julio Ponce and Adem Karahoca, , Vienna, Austria, pp. 438.

[79] Duntsch I. and Gedigal G. (1997) ‘Statistical Evaluation of Rough Set Dependency Analysis’, *International Journal of Human-Computer Studies*, 46(5) pp. 589–604.

[80] Berners-Lee T., Hendler J. And Lassila, O. (2002) ‘The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. *Scientific American Special Online Issue*.

[81] Miller G. A., Beckwith R., Fellbaum C., Gross D. and Miller K. (1993) ‘Introduction to WordNet: An On-line Lexical Database’, Available at: http://wordnetcode.princeton.edu/5papers.pdf.

[82] Janik M. and Kochut K. (2008) ‘Training-less Ontology based Text Categorization’, *In Workshop on Exploiting Semantic Annotations in Information Retrieval (ESAIR 2008): the 30th European Conference on Information Retrieval (ECIR'08)*, March 2008.

[83] Kudelka M., Snasel V., Lehecka O., El-qawasmeh E. and Pokorny J. (2009) ‘Semantic Annotation of Web Pages Using Web Patterns’, *Advanced Internet Based Systems and Applications*, Lecture Notes in Computer Science, 2009, vol. 4879 pp. 280-291.

[84] Sean, L., Lee, S., Rager, D., and Handler, J. (1997) ‘Ontology-based web agents’, *Proceeding of the First Int. Conf. on Autonomous Agents (Agents'97)*, USA, pp. 59-68.

[85] Handschuh, S., Staab, S., Ciravegna, F. (2002) ‘S-CREAM Semi-automatic CREAtion of Metadata’, *the 13th Int. Conf. on Knowledge Engineering and Management* (*EKAW2002*).

[86] Sen A. (2004) ‘Metadata management: past, present and future’, *Decision Support Systems*, 37 (1) pp. 151–173.

[87] Erdmann M., Maedche A., Schnurr H.P. and Staab S. (2001) 'From manual to semi-automatic semantic annotation: About ontology-based text annotation tools', ETAI *Journal - Section on Semantic Web*, 6(2).

[88] Corcho O. (2006) 'Ontology based document annotation: trends and open research problems', International Journal of Metadata, Semantics and Ontologies, 1(1) pp.47-57.

[89] Wilbur W. J., Sirotkin K. (1992) 'The automatic identification of stop words', *Journal of Information Science*, 18(1) pp. 45-55.

[90] Frakes W. B. (2010) 'Stemming Algorithm', *Software Engineering Guild*, Sterling, [Online] Available at:

http://orion.lcg.ufrj.br/Dr.Dobbs/books/book5/chap08.htm **(Accessed May 2010).**

[91] C. Caragea, C. Caragea, N. McNeese, A. Jaiswal, G. Traylor, H. Kim, P. Mitra, D. Wu, A. H. Tapia, L. Giles, B. J. Jansen and J. Yen (2011) 'Classifying Text Messages for the Haiti Earthquake', *Proceedings of the 8th International ISCRAM Conference*, May 2011, Lisbon, Portugal, pp. 1-10.

[92] Quan X., Liu G., Lu Z., Ni X. and Wenyin L. (2009) 'Short text similarity based on probabilistic topics', *Knowledge and Information Systems*, 25(3) pp. 473-491.

[93] Bloehdorn, S., Hotho, A. (2004) 'Text Classification by Boosting Weak Learners based on Terms and Concepts', the *4th IEEE International Conference on Data Mining* (*ICDM'04*) pp. 331-334.

[94] Zelikovitz, S. and Hirsh, H. (2000) 'Improving Short Text Classification Using Unlabeled Background Knowledge' *Seventeenth International Conference on Machine Learning* (*ICML*), Stanford.

[95] Hotho A., Staab S. and Stumme G. (2003) 'Ontologies improve text document clustering', *3rd Third IEEE International Conference on Data Mining* (*ICDM*

2003).

[96] Rosso P., Ferretti E., Jiménez D. and Vidal V. (2004) 'Text Categorization and Information Retrieval Using WordNet Senses', *2nd Global WordNet International Conference*, GWN-2004, Brno, Czech Republic.

[97] Abdalgader K. and Skabar A. (2010) 'Short-Text Similarity Measurement Using Word Sense Disambiguation and Synonym Expansion', *AI2010: Advances in Artificial Intelligence*, vol. 6464, pp. 435-444

[98] Zesch T., Gurevych I. (2007) 'Analysis of the Wikipedia Category Graph for NLP Application', *TextGraphs-2: Graph-based Methods for Natural Language Processing- HLT-NAACL 2007 Conference*, Rochester, New York, pp.1-9.

[99] Gabrilovich, E., Markovitch, S. (2006) 'Overcoming the Brittleness Bottleneck using Wikipedia: Enhancing Text Categorization with Encyclopedic Knowledge', The *21th National Conference on Artificial Intelligence*, Boston, MA, USA, pp.1301-1306.

[100] Wikipedia: The Free Encyclopedia. Wikimedia Foundation.: [Online] Available at: http://en.wikipedia.org (Accessed: July 2011).

[101] Auer S. and Lehmann J. (2007) 'What have Innsbruck and Leipzig in common? Extracting Semantics from Wiki Content', *European Semantic Web Conference (ESWC'07)*. Springer, Innsbruck, Austria, pp. 503-517.

[102] Krotzsch M., Vradecic D., Volkel M., Haller H. and Studer, R. (2006) ' Semantic Wikipedia', *Web Semantics: Science, Services and Agents on the World Wide Web, 15th International Conference on World Wide Wed (WWW 2006)*, 5(4) pp. 251–261.

[103] Hu X., Sun N., Zhang C. and Chua T. S (2009) 'Exploiting internal and external semantics for the clustering of short texts using world knowledge', *Proceeding of .CIKM*, Hong Kong, China, November 2009, pp. 919-928.

[104] Nagarajan M., Sheth A., Aguilera M., Keeton K. (2006) ’Altering Document Term Vectors for Classification - Ontologies as Expectations of Co-occurrence’, *WWW 2007,* 8–12 May 2007, Banff, Alberta, Canada, pp. 1225-1226.

[105] Hammond B., Sheth A.P. and Kochut, K.J. (2002) ‘Semantic Enhancement Engine: A Modular Document Enhancement Platform for Semantic Applications over Heterogeneous Content’, Real *World Semantic Web Applications*, IOS Press, pp. 29-49.

[106] Sheth A. , Bertram C. , Avant D., Hammond B., Kochut K. and Warke Y. (2002) ‘Semantic Content Management for the Web’, *IEEE Internet Computing*, August 2002, pp. 80-87.

[107] Wu S. H., Tsai T., Hsu W. (2003) ‘Text categorization using automatically acquired domain ontology’, *6th international workshop on Information retrieval with Asian languages*, Sapporo, Japan, vol. 11, pp. 138-145.

[108] Cormack G., Hidalgo J. M. and Sanz E. P. (2007) Spam Filtering for Short Messages, CIKM *2007 Proceedings of the sixteenth ACM conference on Conference on information and knowledge management,* NewYork, USA, pp. 313-320.

[109] Nesic S., Jazayeri M., Crestani F. and Gasevic D. (2010) ‘Concept-Based Semantic Annotation, Indexing and Retrieval of Office-Like Document Units’, *RIAO '10 Adaptivity, Personalization and Fusion of Heterogeneous Information*, Paris, France, pp. 134-135.

[110] Kiryakov A, Popov B., Terziev I., Manov D. and Ognyanoff D. (2004) ‘Semantic annotation, indexing, and retrieval’, *Web Semantics: Science, Services and Agents on the World Wide Web*, 2(1) pp. 49-79.

[111] Nguyen N. V., Ogier J., Tabbone S. and Boucher A. (2009) ‘Text Retrieval Relevance Feedback Techniques for Bag of Words Model in’, *World*

Academy of Science, Engineering and Technology,54, pp. 541-546.

[112] Sriram B., Fuhry D., Demir E., Ferhatosmanoglu H. and Demirbas M. (2010) 'Short text classification in twitter to improve information filtering', *Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval - SIGIR '10*, pp. 841.

[113] Daconta, M. C., Obrst, L. J. and Smith, K. T. (2003) 'The Semantic Web: a guide to the future of XML, Web services, and knowledge management', Indianapolis, Ind.: Wiley Pub.

[114] NISO Press (2004) 'Understanding Metadata', [Online] Available at: www.niso.org/standards/resources/UnderstandingMetadata.pdf (Accessed: August 2008).

[115] Corcho, O. (2000) 'Ontology based document annotation: trends and open research problems', *Int. J. Metadata, Semantics and Ontologies,* 1(1) pp. 47-57.

[116] Extensible Markup Language, [Online] Available at: http://www.w3.org/XML/ (Accessed: August 2011).

[117] Knudsen J. () 'Parsing XML in J2ME', Oracle, SUN Developer Network, [Online] Available at:

http://developers.sun.com/mobility/midp/articles/parsingxml/ (Accessed: September 2008).

[118] W3C Recommendation (2004) 'OWL Web Ontology Language Overview', [Online] Available at: http://www.w3.org/TR/owl-features/ (Accessed: May 2010).

[119] W3C Recommendation (2009) 'OWL 2 Web Ontology Language Document Overview', [Online] Available at: http://www.w3.org/TR/owl2-overview/ (Accessed: March 2011).

[120] Heflin J. ‘ An Introduction to the OWL Web Ontology Language,’ Lehigh University, [Online] Available at

http://www.cse.lehigh.edu/~heflin/IntroToOWL.pdf (Accessed: March 20011).

[121] Reeve L. and Han, H. (2005) ‘Survey of semantic annotation platforms’ *Proceedings of the 2005 ACM symposium on Applied computing - SAC ’05*, pp.1634.

[122] Phan X., Nguyen M. and Horiguchi S. (2008) ‘Learning to classify short and sparse text and web with hidden topics from large-scale data collections’, *Proceedings of the 17th international conference on World WideWeb. ACMPress*, New York, pp. 91–100.

[123] Li Y., McLean D., Bandar Z., et al (2006) ‘Sentence similarity based on semantic nets and corpus statistics’, *IEEE Transaction Knowledge Data Engineering 18*, pp.1138-1150.

[124] Mihalcea R., Corley C. and Strapparava C. (2006) ‘Corpus-based and Knowledge-based Measures of Text Semantic Similarity’, 21st *National Conference on Art. International*, vol. 1, pp.775-780.

[125] Ramage D., Rafferty A. and Manning C (2009) ‘Random Walks for Text Semantic Similarity’, *In ACL-IJCNLP 2009*, pp.2331.

[126] Islam A. and Inkpen D. (2008) ‘Semantic Text Similarity using Corpus-based Word Similarity and String Similarity’, *ACM Transaction on KDD, 2(2)*, pp. 1-25.

[127] Okazaki N., Matsuo Y., Matsumura N. et al (2003) ‘Sentence extraction by spreading activation through sentence similarity’, IEICE *Transaction Information System E86D (9)* pp.1686-1694.

[128] William C., Text Classification, Tutorial, [Online] Available at:

http://www.clairlib.org/index.php/Text_Classification_Tutorial (Accessed: Febrauary 2011).

[129] Yang, Y.(1994) 'Expert network: Effective and efficient learning from human decisions in text categorization and retrieval', *17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New York, USA, pp.13-22.

[130] Sriram, B. (2010) 'Short text classification in twitter to improve information filtering', Master dissertation, The Ohio State University.

[131] Huaxiong L., Xianzhong,Z and H. Bing (2009) 'Method to determine α in rough set model based on connection degree', 20(1) pp. 98-105.

[132] Garruzzo S. and Rosaci D. (2008) 'Agent clustering based on semantic negotiation', *ACM Transactions on Autonomous and Adaptive Systems*, 3(2) pp. 1-40.

[133] Al-Sultany G., Li M., Jan S. and Al-raweshidy H. (2010) 'Facilitating Mobile Communication with Annotated Messages', *the 10th IEEE International Conference on Computer and Information Technology*, Bradford, UK, pp. 755-760.

[134] Grinstead C. M. and Snell J. L. ' Introduction to Probability' , Available at: http://www.dartmouth.edu/~chance/teaching_aids/books_articles/probability_book.

[135] Wang X. H., Zhang D. Q., Gu T. and Pung H. K. (2004) 'Ontology Based Context Modeling and Reasoning using OWL', *Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second IEEE Annual Conference*, 14-17 March 2004, pp. 18-22.

[136] McGuinness D. L. and Harmelen F. (2004) 'OWL Web Ontology Language Overview', *World Wide Web Consortium (W3C) recommendation*, February 2004. Available at:

http://cies.hhu.edu.cn/pweb/~zhuoming/teachings/MOD/N4/Readings/5.3-B1.pdf

[137] Horridge M., Simon J., Georgina M., Alan R., Robert S., and Chris W. (2007) ‘A Practical Guide To Building OWL Ontologies Using Protégé 4 and CO-ODE Tools’, , *The University of Manchester*, Edition 1.1.

[138] Liu H., Xiong S. and Fang Z (2011) ‘FL-GrCCA: A granular computing classification algorithm based on fuzzy lat tices’, *Computers & Mathematics with Applications*, 61(1) pp.138-147.

[139] Zainal A., Maarof M. A. and Shamsuddin S. M. (2006) 'Feature Selection Using Rough Set in Intrusion Detection’, *TENCON 2006 IEEE Region 10 Conference* 14-17 November 2006, Hong Kong, pp. 1-4.

[140] Suresh, G.V., Reddy E.S., Shabbeer S. S. (2011) ‘Rough set analysis for uncertain data classification’, *Trendz in Information Sciences and Computing (TISC), 2011 3rd International Conference*, Chennai , pp. 22 – 29.

[141] Jensen R. and Shen Q. (2003) ‘Finding Rough Set Reducts with Ant Colony Optimization’, Proceeding 2003 UK Workshop Computational Intelligence, pp. 15-22.

[142] Large Scale Distributed Information Systems Lab (LSDIS) Department of Computer Science, University of Georgia 410 Boyd Graduate Studies Research Center, Athens, GA 30602-7404

[143] Antonie, M. L. (2003) ‘Text Document Categorization by Term Association’, *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference*, 10 March 2003, pp.19 – 26.

[144] The text classification problem, Available at: http://nlp.stanford.edu/IR-book/html/htmledition/the-text-classification-problem-1.html.

[145] Jan, S., Li, M. and Al-Sultany, G., Al-Raweshidy, H. (2010) 'File annotation and sharing on low-end mobile devices', *Fuzzy Systems and Knowledge Discovery (FSKD), 2010 Seventh International Conference*, Yantai, Shandong , 10-12 August 2010, pp.2973-2977.

[146] Narli S. (2010) 'An alternative evaluation method for Likert type attitude scales: Rough set data analysis', Scientific Research and Essays, Academic Journals, 5(6) pp. 519-528, Available at: http://www.academicjournals.org/SRE .