Ontology Learning from the Arabic Text of the Qur'an:

Concepts Identification and Hierarchical Relationships Extraction

Sameer Mabrouk A. Alrehaili

Submitted in accordance with the requirements for the degree of
Doctor of Philosophy

The University of Leeds
School of Computing

November, 2017

The candidate confirms that the work submitted is his/her own, except where work which has formed part of jointly-authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

Publications

Some of the work presented in chapters Chapter 3, 4, 5, 6, and 7 of this thesis are based on jointly-authored publications. The candidate is the principal author of all original contributions presented in these papers, the co-authors acted in an advisory capacity, providing feedback, general guidance and comments. All original contributions presented here are my own.

Chapter 3

S. M. Alrehaili and E. Atwell, "Computational ontologies for semantic tagging of the Quran: A survey of past approaches," in *Proceedings of the 2nd Workshop on Language Resources and Evaluation for Religious Texts*, 2014, pp. 19–23.

Chapter 4

S. M. Alrehaili and E. Atwell, "Discovering Qur'anic Knowledge through AQD: Arabic Qur'anic Database, a Multiple Resources Annotation-level Search," in *2nd IEEE International Workshop on Arabic & derived Script Analysis and Recognition (ASAR)*, 2018, pp. 102–107.

S. M. Alrehaili, M. Alqahtani, and E. Atwell, "A Hybrid Methods of Aligning Arabic Qur'anic Semantic Resources," in *2nd IEEE International Workshop on Arabic & derived Script Analysis and Recognition (ASAR)*, 2018, pp. 108–113.

I carried out the design, implementation and evaluation of all computational work related to structure-based and the hybrid-based method. I wrote the majority of the content of the paper. Mohammed Alqahtani implemented the fuzzy lexical-based method. Eric Atwell provided supervisory advice. they also made suggestions to help clarify the content for submission.

Chapter 5

S. M. Alrehaili and E. Atwell, "Extraction of Multi-Word Terms and Complex Terms from the Classical Arabic Text of the Quran," International Journal on Islamic Applications in Computer Science And Technology, vol. 5, no. 3, pp. 15–27, 2017.

Chapter 6

S. M. Alrehaili and E. Atwell, “Concepts and Concept Hierarchies Extraction from Arabic text of the Quran, ”.

Acknowledgements

First and foremost, thank you Allah (Almighty God) for giving me the strength, knowledge, wellbeing, and ability to undertake some of the most important work and experience in my life. Thank you for giving me the opportunity to work with your words (the holy Qur'an) leveraging on the advancement of computational semantics. Without your blessings, everything would have not been possible in my life, including this thesis. Reward my effort for doing this thesis.

I dedicate this thesis to my family who have always supported me. My deepest gratitude goes to my parents, Mr Mabrouk and Mrs Rajwah, whose dreams for me have resulted in this achievement and with for their love, support, motivation, prayers and care. They have always supported my education and have been a great motivation to accomplish this degree. I would also thank them for their patience and waiting for me until I finish my PhD when I was studying abroad.

I am grateful to my supervisor, Dr. Eric Atwell, for his support, and encouraging comments throughout the duration of my PhD. Dr Atwell's continuous encouragement enabled me to gain a great learning experience in several dimensions to make my work publishable.

To my wife, Ahlam and my children Mayar, Mohammed and Muadh for their love, patience and support during my study. To my brothers and sisters in Saudi Arabia (Aminah, Amal, Abdullah, Sami,Osama, Hammam, Arwa and Yousef) who were always beside me to support me. I appreciate all that you did for me. This achievement would not have been possible without their support and encouragement.

I would also like to acknowledge the hard work of my colleagues in the NLP group for sharing ideas and working together, discussing my thoughts, answering my numerous questions and even cooperating successfully many times. Thank to Dr. Abdullah Alfaifi, Dr. Ahmed Alzahrani, Mohammad Alqahtani, Abdulrahman Alosaimy, Ayman Alghamdi. Thanks for all annotators for their useful work in annotating domain-specific terms for the Qur'an sample chapter no 29. Thanks for all reviewers for their insightful comments on the published papers.

Finally, I owe special thanks to the Saudi Arabian Ministry of Higher Education for granting me a scholarship to pursue my studies in the UK. I especially want to thank the King Abdullah scholarships program for awarding me a scholarship for the first two years of my PhD research and the College of Computer Science and Engineering, Yanbu, Taibah University for awarded me with the scholarship to carry out the last two years.

Abstract

Recent developments in ontology learning have highlighted the growing role ontologies play in linguistic and computational research areas such as language teaching and natural language processing. The ever-growing availability of annotations for the Qur'an text has made the acquisition of the ontological knowledge promising. However, the availability of resources and tools for Arabic ontology is not comparable with other languages. Manual ontology development is labour-intensive, time-consuming and it requires knowledge and skills of domain experts.

This thesis aims to develop new methods for Ontology learning from the Arabic text of the Qur'an, including concepts identification and hierarchical relationships extraction. The thesis presents a methodology for reducing human intervention in building ontology from Classical Arabic Language of the Qur'an text. The set of concepts, which is a crucial step in ontology learning, was generated based on a set of patterns made of lexical and inflectional information. The concepts were identified based on adapted weighting schema that exploit a combination of knowledge to learn the relevance degree of a term. Statistical, domain-specific knowledge and internal information of Multi-Word Terms (MWTs) were combined to learn the relevance of generated terms. This methodology which represents the major contribution of the thesis was experimentally investigated using different terms generation methods. As a result, we provided the Arabic Qur'anic Terms (AQT) as a training resource for machine learning based term extraction.

This thesis also introduces a new approach for hierarchical relations extraction from Arabic text of the Qur'an. A set of hierarchical relations occurring between identified concepts are extracted based on hybrid methods including head-modifier, set of markers for copula construct in Arabic text, referents. We also compared a number of ontology alignment methods for matching ontological bilingual Qur'anic resources.

In addition, a multi-dimensional resource named Arabic Qur'anic Database (AQD) about the Qur'an is made for Arabic computational researchers, allowing regular expression query search over the included annotations. The search tool was successfully applied to find instances for a given complex rule made of different combined resources.

بِسْمِ ٱللَّهِ ٱلرَّحْمَٰنِ ٱلرَّحِيمِ

وَإِن تَعُدُّوا۟ نِعْمَةَ ٱللَّهِ لَا تُحْصُوهَآ ۗ إِنَّ ٱللَّهَ لَغَفُورٌ رَّحِيمٌ ١٨

"And if you should count the favors of Allah , you could not enumerate them. Indeed, Allah is Forgiving and Merciful."

The holy Qur'an, verse (16:18)

Contents

List of Tables

List of Algorithms

List of Figures

List of Abbreviations

The following glossary lists all acronyms used throughout this thesis with their definition. The page number indicates the abbreviation first use in the thesis.

	-		
ATR	-	Automatic Term Recognition	16
MWTs	-	Multi-word Terms	vi
AQT	-	The Arabic Qur'anic Terms	vi
AQD	-	The Arabic Qur'anic Database	vi
TE	-	Term Extraction	16
NLP	-	Natural Language Processing	1
OL	-	Ontology Learning	1

Part I
Introduction, Background and Literature Review

Chapter 1

Introduction

As humans, we understand a sentence immediately without noticing that the process of understanding of what we read or hear is very complex and requires a sophisticated mechanism in order to get the meaning for given utterances. For computers, it is not a trivial task to learn or understand natural languages. A computer system has to deal with all of the aspects of language (i.e. tokenisation, lexical and syntax analysis, semantic and inference) in order to the meaning of what is written.

Natural Language Processing (NLP) teams from universities around the world have addressed the issue of knowledge representation and its impact on improving the development of intelligent systems for the Qur'an [1]. Computational Semantics and Computational Linguistics work together to provide semantic representations for natural language. NLP provides methods and tools to uncover knowledge representation through linguistic cues [2], while Computational Semantic provides the formality and allows tasks like reasoning for this representation [2]. These representations allow computers to understand the text to acquire knowledge, make inferences, reasoning, answer given questions, and retrieve relevant information for a given query. One of the most common semantic representations for knowledge is called ontology, which is one of the main components related to this thesis topic. The following section gives a brief background to the topic of the thesis.

1.1 Ontology and Ontology Learning

The topic of this thesis is ontology learning (OL) from the Arabic text of the holy Qur'an. Ontology is defined as "an explicit specification of a conceptualization", which is the most widely cited definition of ontology, by [3]. Broadly speaking, the purpose of ontology in computer science is to represent a common domain understanding [4]. An ontology comprises of individuals, terms, and relations, which are used to describe the domain of interest in an explicit way. Figure 1.1, Figure 1.2 and Figure 1.3 depict the scope of angels in the Qur'an from three different ontological annotations of the Qur'an.

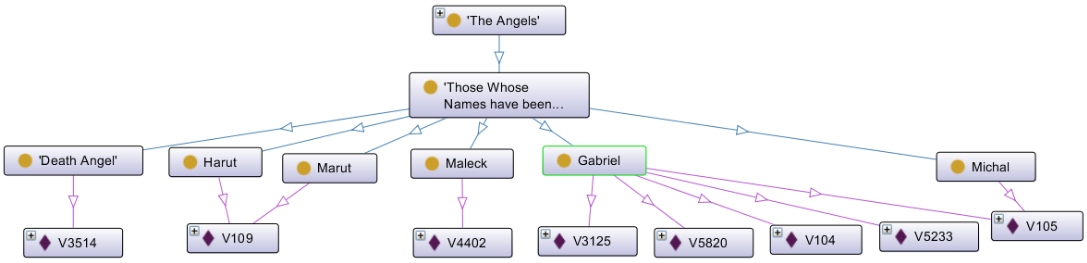


Figure 1.1 Angels from the Qurany ontology

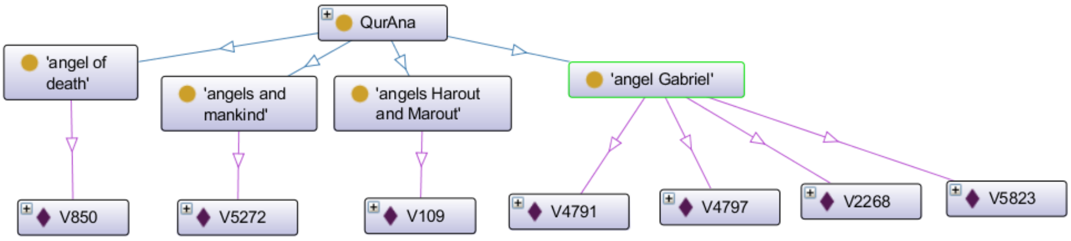


Figure 1.2 Angels from the QurAna ontology

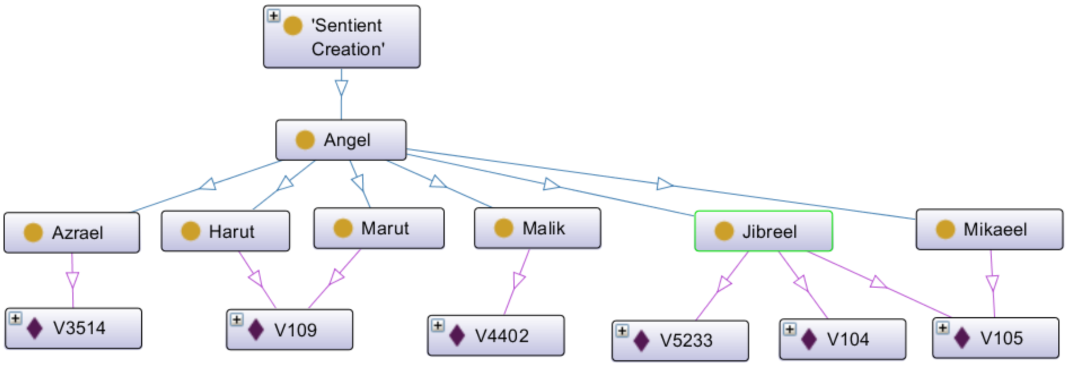


Figure 1.3 Angels from the QAC ontology

In the lowest level of these ontologies, the purple diamonds denote the individual verses which are linked to a concept for representing the occurrences of that concept in relation to the Qur'an. The yellows circles are the terms of the ontology. The highest level are called abstract concepts and the lowest are called concrete concepts. The blue arrows denote to "is-a" relationships between concepts, while the purples denote the instances between a concept and where it has mentioned in the text. In these ontologies, the instances are the verses where the actual instances are mentioned. Both ontologies Qurany and QAC have some common verses and concepts. While QurAna's instances are different because it represents the occurrences of the pronouns of these concepts. For example, the concept "Gabriel/Jibreel" mentioned directly in a number of verses such as "v105", "v104" and "v5233" according to Qurany and QAC. However Qurany has more occurrences than QAC. While QurAna occurrences include some not directly mentioned in the Qur'an

Thesis topic also includes the term "ontology learning", which refers to the way of reducing human intervention, labour-intensive and time-consuming work in the process of ontology construction using automatic or semi-automatic methods of ontology construction. This problem is known as the "knowledge acquisition bottleneck" [5]. Although ontology learning from text aims to automate the process of ontology construction, in most cases results still need to be validated and modified by humans before using them in applications [6]. However, the results can be evaluated with respect to a similar resource in order to automate the validation and to assess the quality of learned ontology. This way of evaluation is known as a gold-standard. Automatic validation is not always available, this depends on how the resources of the targeted domain available are. The domain of Arabic language needs more work to produce such resources in order to automate the evaluation of ontology construction [7][8][9].

1.2 Why convert text to ontologies

Texts often carry stabilized and shared knowledge by communities of practice [10]. Furthermore, domain experts are not always available to participate in ontology construction and texts are more readily available than experts [10]. Ontologies are very popular for modelling domain knowledge among various domain fields. They provide a set of standard notations that supports formality, explicitness, sharing, reusing, they also can be queried for retrieving related knowledge and they can be reasoned with for inferring new relations. Ontologies also play an important role in different broad disciplines and their use and applications have attracted many researchers from different areas. Therefore, any attempt for automating the transformation of a textual source from a particular domain to such a model would add several benefits to the domain.

Ontologies are also important for sharing the common understanding of the text among researchers and software agents [3]. Moreover, domain knowledge can be reused by other domains [3]. For example, the properties and metadata of date and time are being used in several domains. Furthermore, ontologies are useful in order to make domain assumptions explicit [11]. In addition, ontologies play an important role to separate domain knowledge from operational knowledge [11]. Ontologies also make use of domain knowledge analysis [11].

1.3 Motivation and Aim

The Qur’an comprises the divine words of wisdom considered and used as the prime source of knowledge and guidance for Muslims throughout the world for fourteen centuries. Different kinds of knowledge captured in the Qur’an can be extracted such as how to deal with daily life basics including marriage, divorce, children’s rights, parental rights, etc. Many different types of annotations are available, which is one reason why we choose the text of the Qur’an. Therefore, we will not start from the scratch. It has been studied for more than 1400 years. The Qur’an is also considered as the main source for grammar in Classical Arabic and it is being used to explain grammatical phenomena in the Arabic language, therefore it is the optimum representative for Classical Arabic text [12]. It is also the purist and the most authentic resource from the classical Arabic language [13].

Usually, an ontology is constructed to cover a domain of interest using several resources and books that cover the domain of interest. This research uses the Arabic text of the Qur'an and other Qur'anic annotations for extracting the main ontological knowledge; concepts and hierarchical relations. Using actual text instead of a predefined list from another resource may represent the reality of the domain. Most of existing research on ontology learning for Arabic text focuses on mapping or merging available annotations rather than learning from actual text or using these annotations as external resources. The Arabic language has some of culture-specific terms that have no equivalents in English [14]. A number of evidence and examples discussed for non-equivalence in the process of translating from Arabic to English can be found in [14].

The increasing usage and development of Qur'anic computing recently was the main inspiration behind this thesis research. These uses have allowed this type of ontology to play a crucial role in many different intelligent systems such as Question Answering. However, little prior work has been done for identifying Qur'anic concepts and relations using ontology learning principles. The lack of ontological extraction methods that consider the specifications of Arabic language has limited the research in this field for the Arabic language. The lack of sufficient semantic sources for the Arabic language is another reason.

A number of Ontology learning components for Arabic language have to be provided such as concept identification and hierarchical extraction since this language has different features than other languages. These features have a reflection on the terms and relationships in Arabic ontology. For example, highly complex inflection makes terms more difficult to generate. Some of these difficulties led other researchers on Arabic ontology development to adopt a manual approach.

There has been an increase on research in Semantic annotation or representation in the wider area including Islamic knowledge. For example, [15], [16], [17], [18], [19], [20], [21], [22] used ontology in their research for representing Islamic knowledge. Ontology learning is a central for many different applications. The aim of this thesis is to reduce human intervention (time-consuming and labour-intensive work) in ontology construction from the Arabic text of the Qur'an. We concentrate on developing new methods for Ontology learning from the Arabic text of the Qur'an, including concepts identification and hierarchical relationships extraction.

1.4 Objectives

1. To review existing methodologies for ontology learning from textual resource with a focus on concept and conceptual relationship extraction.
2. To review the existing Qur'anic ontologies according to a range of appropriate criteria.

   A comparison of recent Qur'anic ontology projects based on 9 criteria including language, coverage area, coverage proportion, underlying format, underlying techniques, availability, number of concepts, number of relations and types and verification method.
3. To develop a method for concept identification from the Classical Arabic text of the Qur'an.
4. To find hierarchical relations (is-a) among identified concepts.

1.5 Thesis Contributions

A new Qur'anic database, AQD, that combines five different ontological and other annotations for the Qur'an. This resource is attached to an annotation-based search, which allows users to extract knowledge based on a query that can contain several types of features. Such an environment will enable researchers in Arabic computational linguistics and Qur'anic research to investigate new directions and new features for different NLP learning tasks.

- A new framework for Domain-Specific Terms of the Qur'an and concepts.

  The proposed framework takes into consideration the importance of morphological features that detect the lexical stable unit. In addition, it combines domain-specific knowledge and statistical knowledge for measuring the importance of a term. To our knowledge, this work is the first that exploits inflectional, domain-specific knowledge and statistical knowledge in concept identification for the ontology learning task. A new standard resource for research on Classical Arabic term extraction is provided as a dataset which contains 10351 domain-specific terms validated manually. We named this dataset as Arabic Qur'anic Terms (AQT).

- An original approach of automatic hierarchical relations construction based on a combination of head-modifier and a three markers for a linguistic construct called copula. We show its success in constructing the hierarchy between identified concepts for some domains such as names of Allah. To the best of our knowledge, this is the first work proposes a method for extracting inexplicit "is-a" relations for the Arabic language based on pronominal information. And presenting a hybrid methods combine Head-Modifier and a number of Copulas patterns for extracting is-a relationships from Arabic text. We encoded three previous Qur'anic ontological annotations (Qurany, QurAna and QAC) using Web Ontology Language (OWL).

- A new hybrid ontology alignment which is aggregating multiple similarity measures for a given pair of concept. It takes advantage of combining fuzzy bilingual lexical and structure based methods.

- A supervised model for learning implemented in an experiment for Arabic Qur'an term extraction base on AQT dataset.

1.6 Outline of the thesis

This thesis is split into four parts with 8 chapters as shown in Figure 1.4.

- **Part I**

  Introduction, Background and Literature Review

  - Chapter 1 Introduction
  - Chapter 2 Background
  - Chapter 3 Literature Review and Related Work

- **Part II**

  Collecting and Discovering Qur'anic Resources

  - Chapter 4 Combining and Extracting Qur'anic Knowledge: The AQD Arabic Qur'anic Database

- **Part III**

  Modelling Qur'anic Ontology Learning

  - Chapter 5 Concept Identification
  - Chapter 6 Hierarchical Structure

- **Part IV**

  Review, Evaluation, Future Work and Conclusion

  - **Error! Reference source not found.** Review and Evaluation
  - Chapter 8 Future Work and Conclusions

Figure 1.4 The structure of the thesis

- **Part I** includes three chapters; introduction, background and the literature review.
  - **Chapter 2** provides background information of the related topics and methods that are going to be used such as Qur'anic

    - domain, ontology, and some methodologies from Natural Language Processing.
    - **Chapter 3** positions our work in relation to current and past work within the area of ontology learning and the Arabic and Qur'anic domain. It outlines and discusses relevant approaches that have been applied in similar work, focusing on the domain of the Arabic text of the Qur'an, ontology learning from textual resources with focus on concept and hierarchical relation extraction.
- **Part II** includes chapter 4.
    - **Chapter 4** This chapter produces a multi-dimensional resource named Arabic Qur'anic Database (AQD) for combining 5 different ontologies and annotations of the Qur'an. In addition it provides a regular expression like query system for extracting complex grammatical and ontological instances from the AQD.
- **Part III** includes two chapters; 5 and 6.
    - **Chapter 5** provides a framework for generating terms and using weighing schema based on multiple information for measuring their relevance. In addition, a number of experiments based on the method of terms extraction and the method of evaluation are presented. An experiment in applying machine learning algorithms has been conducted for learning Qur'anic domain-specific terms. In this chapter, we exploit the resource AQT which was manually validated to build a model using a machine learning algorithm.
    - **Chapter 6** introduces a new approach for hierarchical relations from the Arabic text of the Qur'an. A set of hierarchical relations occurring between identified concepts are extracted based on hybrid methods including head-modifier, set of rules for copula construct in Arabic, referents, derivational information and alignment with existing ontologies.
- **Part IV** includes two chapters; 7 and 8.
    - Error! Reference source not found. presents a critical review and evaluation for the presented work in the thesis.

    - **Chapter 8** outlines the thesis aims, achievements, limitations, suggestion for future work and conclusions.

Chapter 2

Background

This chapter gives a background information on the topics of the thesis. Firstly, it gives an overview of the Qur'an for those who have not heard about it before. Secondly, it focuses on the terminology of ontology and its contents, paying more attention on ontology that formed from textual resources. Moreover, this chapter defines and explain terms and methods used later in this thesis.

2.1 The Holy Qur'an

The Holy Qur'an is the last sacred book among books that were sent down to God's prophets, peace be upon them all. The Holy Qur'an is undoubtedly an important book; Muslims take the rules and guidance from the Qur'an such as rules of marriage, divorce, inheritance, finance, etc. The structure of the Qur'an is different from the traditional book structure. The Holy Qura'n is composed of verses, also known as "Aya" (plural: Ayat); there are 6236 verses in the Qur'an, categorised in 114 Chapters known as "Sura". A verse is the shortest division in the Qur'an and is a group of words that is complete in itself. Chapters vary in length; one chapter has 286 verses while another has only 3 verses. This division into Sura and Aya helps in referring to a specific verse, the notation (113:1) means we refer to a chapter (Sura) number 113 and within this, verse (Aya) number 1. Part or "Juz" is another type of division, where a group of chapters and verses is assigned to a specific part. Each part has a number of division called “Hizb”, which groups related verses together.

The Holy Qur’an was sent down through the Holy Spirit (angel Gabriel) to the prophet Muhammad during a period of approximately 23 years from 610 to 633 CE [23]. The Holy Qur’an was not sent down as a single book as it is known today; neither was it revealed in a single session. The revelation came in response to specific events. Therefore, in order to understand the Qur’an it is important to know about the prophet Muhammad’s history. The first part of the revelation was in Mecca, the city in which the prophet Muhammad was born.

The Qur’an comprises the divine words of wisdom considered and used as the prime source of knowledge and guidance for Muslims throughout the world for fourteen centuries. Different kinds of knowledge captured in the Qur’an can be extracted such as how to deal with daily life basics such as marriage, divorce, children’s rights, parental rights, how to treat your neighbours and your wife, etc. It has a criminal justice system and also explains the process of the pregnancy and its stages. Many different types of annotations are available, which is one reason why we choose the text of the Qur’an. Therefore, we will not start from the scratch. It has been studied for more than 1400 years. The Qur’an is also considered as the main source for grammar in Classical Arabic and it is being used to explain grammatical phenomena in the Arabic language, therefore it is a good representative for Classical Arabic text. Most of existing research on ontology learning for Arabic text focuses on mapping or merging available annotations rather than learning from actual text or using these annotations as external resources. Using actual text instead of a predefined list from another resource may represent the reality of the domain. More details about the Qur’an can be found in [24].

2.2 What is An Ontology

The term “ontology” was used for a long time in the field of philosophy to refer to the study of existence or the study of being [6]. It was also adopted by several disciplines in Computer Science including Artificial

Intelligence, knowledge representation, logic, and information retrieval. Broadly speaking, the purpose of ontology in computer science is to represent a common understanding of a domain [4]. Ontology is described to be the backbone of the Semantic Web and it has an important role in some semantic applications such as semantic search and Question Answering. The most widely cited definition of ontology, by [3], is "an explicit specification of a conceptualization". The formal specifications allow the common understanding of the domain to be shared among researchers and software agents [3]. Moreover, domain knowledge can be reused and it can reuse other knowledge from other domains [3]. For example, the metadata of the date and time can be reused and shared knowledge in any domain. Furthermore, to make domain assumptions explicit [11]. In addition, to separate domain knowledge from operational knowledge [11]. Ontologies also make a useful of domain knowledge analysis [11].An ontology consists of terms, concepts, relations, and optionally[25] axioms for validation and enforcing constraints [2]. Ontologies are usually described as a directed graph, the nodes of the graph denote the terms or concepts while the edges represent the relations between the nodes [6], [25]. The linguistic representation of a concept is called a term. A concept can be defined as the set of key domain terms or key-phrase according to [26], [27]. For example, in a special domain like "Fruits", most important terms like the "tree" and "banana" could be one of the concepts set, while "root" and "leaves" could be relevant but not as important as these. For ontology learning from text it is common to select the most relevant terms as a set of concepts. The Figure 2.6 in page 23 depicts an example of these ontology contents in a graph.

The process of extracting relevant instances from the data to the concepts of the ontology is called ontology population or knowledge markup [25], [28]. Whereas automatic or semi-automatic support for deriving the concepts and the relations from the data to form an ontology is referred to as ontology learning [28].

2.3 Ontology Learning From Text

Ontology learning aims to automate or semi-automate the process of ontology construction, or in other words reducing the human intervention in ontology construction. The manual ontology construction is always described as a tedious task because of the requirements of time and effort.

The term Ontology Learning (OL) was introduced in 2001 by [29]. The term itself refers to automatic or semi-automatic ontology construction. It aims to reduce human intervention in the ontology construction process through the development of automatic methods for knowledge extraction about a specific domain. Ontology learning often uses methods from Knowledge Acquisition, Machine Leaning and Natural Language Processing, Information Retrieval, Artificial Intelligence, Reasoning and Database Management [5] [30] [31] [32].Ontology Learning can be explained as an Information extraction subtask [33]. Ontology learning seeks to automatically or semi-automatically extract ontological knowledge including terms, concepts, and relations from several forms of data such as unstructured, semi-structured and structured data. Its overall goal is to extract the ontological knowledge with little human intervention.

2.4 Ontology Learning Tasks

As mentioned above, an ontology comprises of concepts and relations that link those concepts. The process of ontology learning can be applied through five steps; terms, concepts, taxonomy, relations and rules or axioms as can be seen in Figure 2.1. In the First step, the natural language terms for the given domain are selected to represent the low level of the ontology. This step is an important especially in ontology learning from text. The second step is required to form the concepts in the next step because one of a set of synonyms can be a concept-term for the rest of them. It is necessary for avoiding redundancy and variation in concepts. Next, is forming the concept as mentioned before. After that, assigning these concepts to others from an upper level ontology. Some methods use Named Entity Recognition techniques to link between concepts and a list of predefine semantic tags such as person, organisation and place. Non-taxonomic relation comes then to enrich ontology relations. The final stage is called rules or axioms which is a list of logic-based rules for inferencing new entities and relationships by enforcing some constraints among entities. For example, if concepts A is a person and has a moustache or beard this implies that A is an adult male.

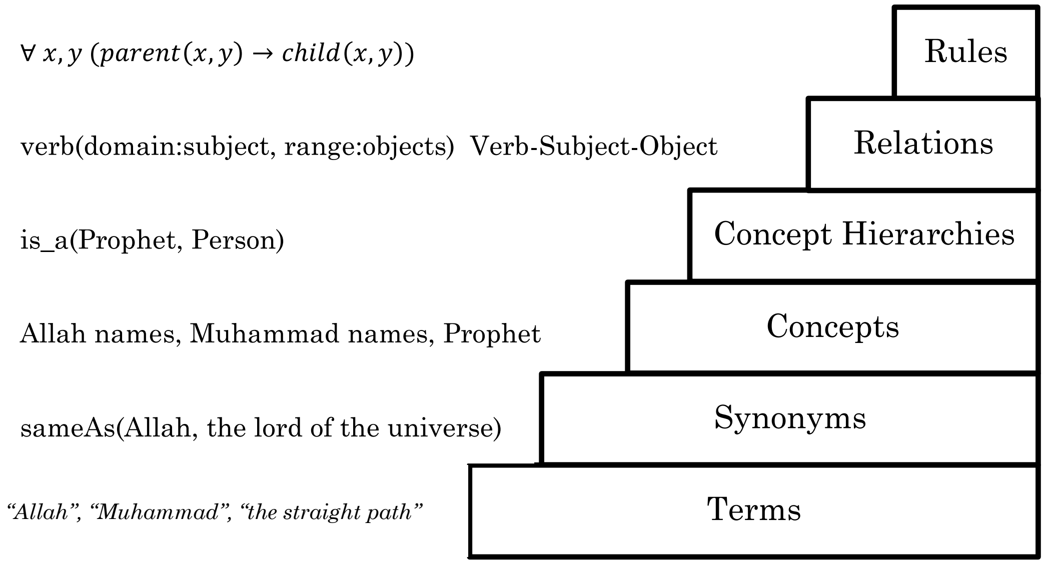

$\forall x, y\ (parent(x, y) \rightarrow child(x, y))$
Rules
verb(domain:subject, range:objects) Verb-Subject-Object
Relations
is_a(Prophet, Person)
Concept Hierarchies
Allah names, Muhammad names, Prophet
Concepts
sameAs(Allah, the lord of the universe)
Synonyms
"Allah", "Muhammad", "the straight path"
Terms

Figure 2.1 Ontology Learning Layer Cake by [28]

2.5 Domain-Specific Terms

An initial step to build a domain ontology is to find the important concepts of that domain. This step is based on the extracted domain-specific terms. Most concept extraction methods execute this in two steps; generate the candidates, then find the important terms. However, generating candidates is not easy as most of domain-specific terms are composed of multi-word terms.

The Qur'an has its own specific terms that should not be missed during ontology construction. The following are a list of term types found in the Qur'an:

2.6 Qur'anic Terms

Like many other domains, terms in the domain of the Qur'an are used to describe entities and objects in the texts. However, a number of variations in Arabic terms make the automatic term extraction a complex task. An estimation of terminology variation amounts to be between 15% to 35% [34] depending on the domain and other factors like type of the text, an important percentage that should not be ignored when generating ontology concepts.

2.6.1 Simple Terms

According to [35] , a term[1] is defined as the basic linguistic unit that describes an entity in a domain, represented as one or more words. Terms can be composed of one word named a single term or a group of words known as multi-word terms. Single-word terms are also known as Simple terms which is defined in ISOcat[2] as a term composed of only one root. Other definitions including in [36] defines the simple term as a term made of single word which is not appropriate for Arabic. In Arabic a word may be morphologically complex and contains more than one meaning-unit, therefore we will work with the definition of ISOcat.

[1] Alternative terms to the "terms" used in formal ontology languages include individual, lexical words, and instance.

[2] A data category registry at http://www.isocat.org/

2.6.2 Complex Terms

Multi-word terms (MWTs) are terms that composed of more than one root. MWTs are believed to be less polysemous than single-word terms [36], [37] and also form the majority of any ontology - approximately 85% of domain-specific terms [38]. Multi-word terms or compound terms are important to be extracted automatically because they are more specific and more descriptive to the meaning than the single-word terms [39]. As it was mentioned above, the majority of domain-specific terms are multi-word terms and the specificity of a multi-word term is higher than the specificity of a single-word term [39]. However, multi-word terms have low appearance or frequency in corpora compared to single-word terms [37], [40] and this may hinder statistical-based extraction approaches. Multi-word terms are classified into two main categories: simple term and complex term. Complex terms are not only those that made of two words or more, but they can be nested within other terms as a modifier. In this thesis, we use the word “terms” to describe the domain-specific terms and the word “concept” for describing the important terms. The process of extracting terms automatically is called term extraction (TE), also known as automatic term recognition (ATR). Terms in ontology may include concrete objects, where their meaning is not ploysmous such as names of people or animals, cars, etc. These can be specific to the text and hence cannot be used to provide general search. A term may be an abstract object which is used to describe an object that has different senses of meaning such as a single word term.

2.6.3 Hidden Terms

These terms are not written based on traditional terms patterns like Noun Noun or Adjective Noun. Two type of hidden terms are found in the Qur’an; Single-word Noun suffixed by personal pronoun, and derived terms.

2.6.4 Suffixed Single-word Terms

When a noun is composed of several morphemes like “يدي, كتابي, ايمانهم” the attached pronoun denotes to the entity that owns, performs. The suffix pronoun is attached to a noun and the noun in this case carry more information. To generate the term, the referent of the pronoun is needed to form the new term. The following example, in Figure 2.2 , shows part of verse (7:2), which includes a pronoun attached to a noun vision. The meaning of the whole word is the disbelievers’ vision, which carries more meaning than taking the noun vision.

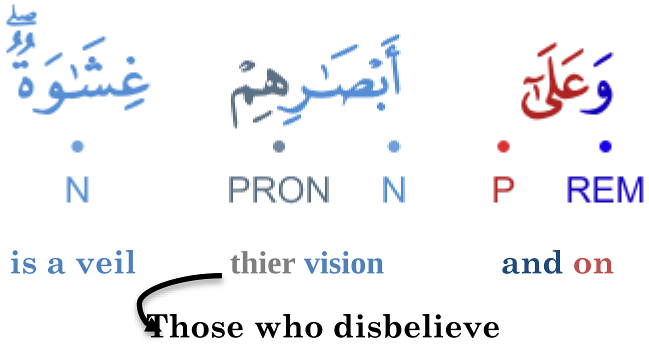


Figure 2.2 An example of a suffix pronoun and their referent information

2.6.5 Derived Terms

A derived term is a term or a concept which was mentioned in verbal form and transformed to the nominal form. Some important Qur’anic terms are missed when adapting term extraction method from other languages due to their limitation to cover specific types of terms. For example, some important terms in the Qur’an like “Iman” and “Purity” were mentioned in verbal form rather than nominal form. Table 2.1 shows four important concepts in the Qur’an were mentioned in verbal more than in nominal form.

Concept	Occurrences	
	V	N
Believe	537	275
Disbelieve	289	202
Remembering	154	139
Purity and washing	17	15

Table 2.1 Example Qur'an concepts which occur in both verbal and nominal forms

Another example is the concept of "Obedience", a very important concept in Islam and the Qur'an. Most Islamic scholars cite verses from the Qur'an to talk about this concept, however these citations are not easily to be understood by computer because they contain this concept in a verbal form. The traditional way of identifying concepts relies on the nominal form while as we mentioned above there are a number of an important set of concepts were mentioned in other forms. The nominal form can be derived from these other forms in order to avoid missing other occurrences of these terms in the text.

2.6.6 Nested Terms

It is a term which starts with a relative noun and considered as a definite noun or nominal phrases in Arabic. For example "alladhīna yu'minūna bil-ghaybi" (Those who believe in the unseen). Table 2.2 shows an example of a complex multi-word term in Arabic that is composed of a syntactic pattern of six different POS tags, found in the QurAna project [41].

Transliteration	mn	lm	yHkm	b	mA	Anzl	Allah
English	whoever	Does not	Judge	By	What	Has revealed	Allah
POS	COND	NEG	V	P	REL	V	PN

Table 2.2 An example of Multi-word term composed of different Part-of-Speech (POS) information

This type of terms also have a multi-word term nested in it, as can be seen in the Figure 2.4 and Figure 2.5 , which show our example of Arabic complex term in two representation models, the term “mA Anzl Allah” is nested in or modifies the main term “mn lm yHkm b mA Anzl Allah”. “Allah” is also another nested term in the nested term of the main term.

A valid term on its own can be part of another term. This term can also be a complex term as it contains more than one root. Figure 2.3 shows the term “māliki yawmi l-dīni” (Sovereign of the Day of Recompense). As can be seen the Day of Recompense is a nested term of the main term and Recompense is also nested in the term Day of Recompense.

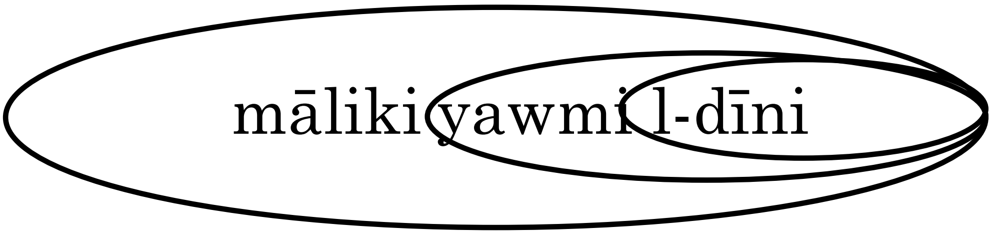


Figure 2.3 A nested term represented by Wilson’s Nested Model [42]

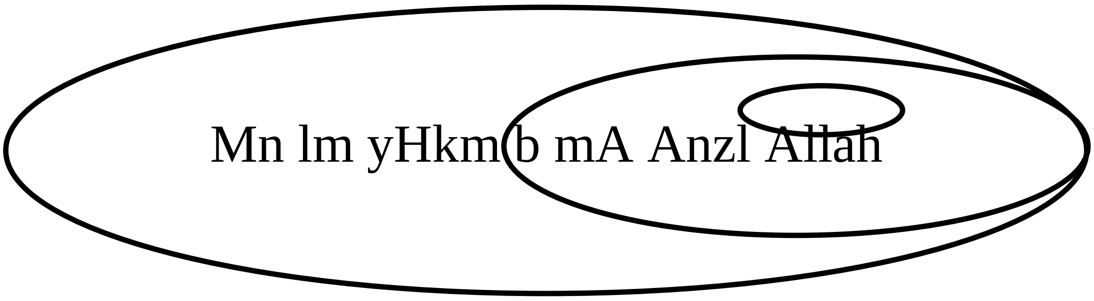


Figure 2.4 Complex term represented by Wilson's Nested Model [42]

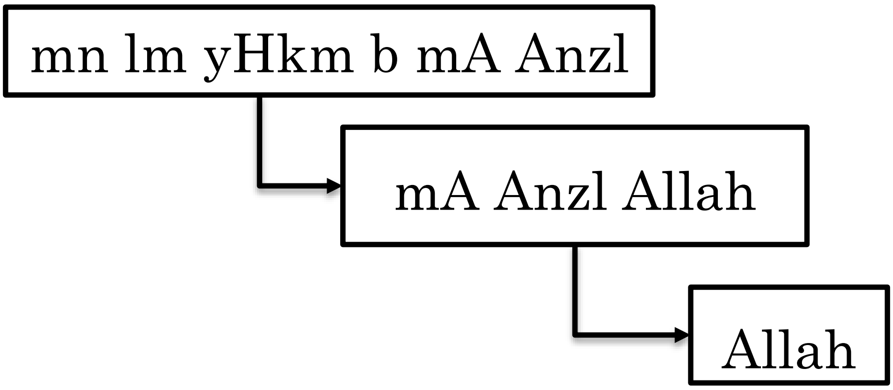


Figure 2.5 Complex term represented by a tree structure

2.7 Qur'anic Terms Variation

A number of linguistic specifications of Arabic multi-word terms can be found [37], [43], [44]. These variations make the automation of Arabic ontology alignment challenging. An example of linguistic variation in Arabic text may occur when comparing a diacriticised concept with and another one is undiacriticised. This may be solved by removing vowels and Hamza from one of compared texts. Another linguistic variation may occur when a concept is being expressed based on varying morphological features. For instance, the concept "The believers" can be expressed in the Qur'an in different ways such as (المؤمنين, المؤمنون) ({lomu&ominiyna,{lomu&ominuwna). Although these different labels of a single concept have been expressed in different lexical word-forms, but they are denoting the same entity. The inflectional feature of case has changed the last two letters of the concept. Another number of morphological features can be changed the way the concept is being expressed such as the determine article and the state. This variation can be solved by matching the two concepts lemmas inflected from the same lemma. Another variation which is special to Arabic Qur'anic text is the type of script. Two different script types, namely Uthamni and Modern Standard Arabic (MSA) are found in the Qur'anic annotations, which have some differences in how words are spelled such as (“الرحمن” “الرحمان”). Another variation which occurs in both Arabic and English translation of the Qur'an is when a written concept is based on its dictionary meaning and not based on its actual word occuring in the Qur'an. An example, from these datasets is (i.e., “الخيل, الحصان”, “الغيث, المطر”, “Gabriel, Jibreel”, “The Gospel, Injeel, The Bible”, “Ibrahim, Abraham”, “Yaqub, Jacob” etc.). Simple normalisation may not help in this case and a more language-dependent method can contribute to an accurate result. Thus, aligning these annotations needs a hybrid approach that takes into account all these variations.

2.8 Ontological Contents and Ontology Formal Languages

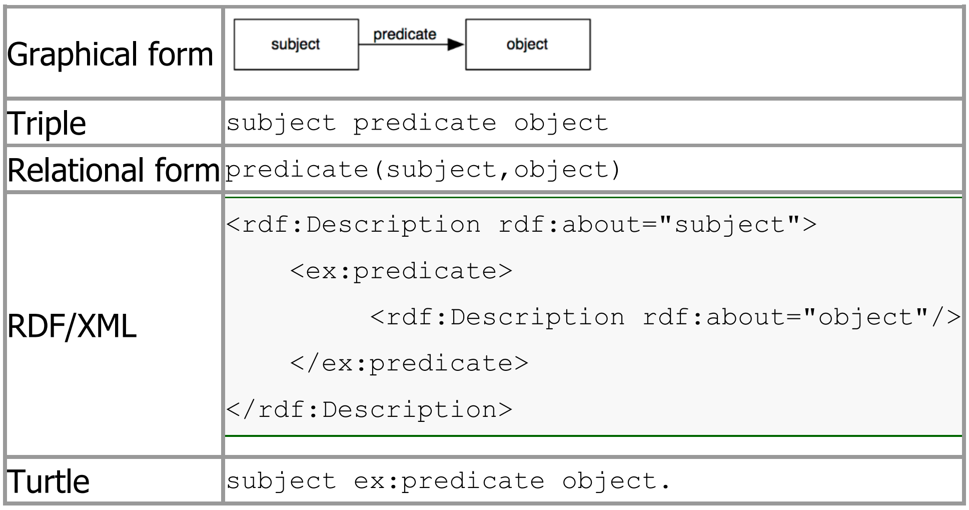
The formalism of Semantic Web contain a number of tools, data model and schemas for describing web resources in a structure manner. This formalism provides a number of notations and vocabularies to represent a domain of interest using these ontology models. There are two types of notations: built-in which is provided by the ontology language; or user-defined which can be defined by the person or the tem who develops the ontology and it can be reused from the same ontology or others.

The most commonly representation of web resources is the Resource Description Framework (RDF), which is a data model for describing resources in a structured manner [45] . RDF schema is used for declaring basic class and types for describing the terms used in RDF. Resources are described by their Uniform Resource Identifiers (URIs), which is a unique identifier for ontological contents, such as classes and relations. RDF document is a graph that composed of a set of triples, the subjects, predicates and objects of these triples are represented as nodes[46]. We will use the Web Ontology Language (OWL) to describe the representation of the basic ontological usage.

The following ‘T’ predicate represents a triple in OWL, which is the simple unit that an ontology made of. The ‘T’ is a directed binary relation between two entities in an ontology.

A triple of the form $T(a, r, b)$

Where: a is an item that denotes the subject in the triple items or the subject of the entity b, r: is the predicate or the property between the two entities, b: is the object that the entity a is related with.



Graphical form	subject —predicate→ object
Triple	`subject predicate object`
Relational form	`predicate(subject,object)`

RDF/XML	`<rdf:Description rdf:about="subject">`
	`<ex:predicate>`
	`<rdf:Description rdf:about="object"/>`
	`</ex:predicate>`
	`</rdf:Description>`

Turtle	`subject ex:predicate object.`

Table 2.3 different forms of RDF triples from [47]

The following example is an example of representing classes and individuals of a topic using OWL. Figure 2.6 shows an example of a hierarchical form of ontology. The ontology in the figure is a part ontology of the Qurany dataset [48], which represents the Qur'anic topics and their relations to the Qur'anic verses. The blue circles represent concepts and sub-concepts while greens represent the instances or individuals, and arrows denote the relationships. "T61" is a concept denote the "Man and The Moral Relations", which has a sub-concept labelled as "Good Morals" and this sub-concept has three sub-concepts; "The Restrain of Anger", "Calmness" and "Be Moderate". Each concept represented in this ontology is labelled with Arabic language and the English translation of the topic. At the leaves level of the ontology the locations of the verses belong to the sub-concepts are linked as individuals. It is obviously shown the purpose of this ontology is to retrieve the relevant verses of the given topic.

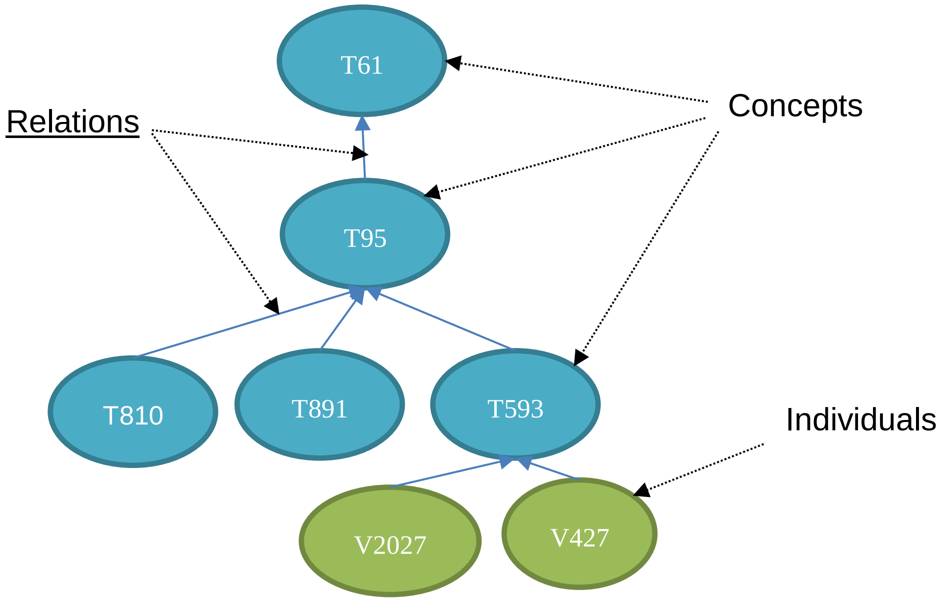


Figure 2.6 The visualisation of classes and their individuals of the Qur'an using Qurany dataset

Concepts and relations from Figure 2.6 can be defined in OWL as in the following:

```
<owl:Class rdf:about="#T61">
        <rdfs:subClassOf rdf:resource="&qur;Topic"/>
        <rdfs:label xml:lang="ar">االإنسان و العلاقات الأخلاقية</rdfs:label>
        <rdfs:label xml:lang="en">Man and The Moral Relations</rdfs:label>
</owl:Class>
<owl:Class rdf:about="#T95">
        <rdfs:subClassOf rdf:resource="#T61"/>
        <rdfs:label xml:lang="ar">الأخلاق الحميدة</rdfs:label>
        <rdfs:label xml:lang="en">Good Morals</rdfs:label>
</owl:Class>
<owl:Class rdf:about="#T539">
        <rdfs:subClassOf rdf:resource="#T95"/>
        <rdfs:label xml:lang="ar">كظم الغيظ</rdfs:label>
        <rdfs:label xml:lang="en">The Restrain of Anger</rdfs:label>
</owl:Class>
<owl:Class rdf:about="#T810">
        <rdfs:subClassOf rdf:resource="#T95"/>
        <rdfs:label xml:lang="ar">السكينة</rdfs:label>
        <rdfs:label xml:lang="en">Calmness</rdfs:label>
</owl:Class>
<owl:Class rdf:about="#T891">
        <rdfs:subClassOf rdf:resource="#T95"/>
        <rdfs:label xml:lang="ar">الاعتدال في الأمور</rdfs:label>
        <rdfs:label xml:lang="en">Be Moderate</rdfs:label>
</owl:Class>
<owl:NamedIndividual rdf:about="#V427">
        <rdf:type rdf:resource="&qur;Verse"/>
        <HasTopic rdf:resource="T539"/>
        <PartOf rdf:resource="C3"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="#V2027">
        <rdf:type rdf:resource="&qur;Verse"/>
        <HasTopic rdf:resource="T539"/>
        <PartOf rdf:resource="C16"/>
        </owl:NamedIndividual>
```

Figure 2.7 The formal representation of Figure 2.6 using OWL

2.8.1 Semantic Relations Among Parts of Text

Text contains parts that are linked to each other in different type of relations, depending on the application that it is going to be used in. For example a paragraph links with a word such that a word “isPartOf” a paragraph. This section concentrates on relations occurring between concepts. A relation can have an inverse such as the inverse relation of “isPartOf” and “hasPart”. It can be transitive that such that their properties can be inherited by classes that links to the subclasses of a class. The following example shows is-a relation between $c_1 \, and \, c_3$ is inherited because c_3 is a sub class of c_2 which is a subclass of c_1.

$$\forall c_1, c_2, c_3 \;\; T(c_1, rdfs{:}\, subClassOf, c_2) \wedge T(c_2, rdfs{:}\, subClassOf, c_3) \\ \Rightarrow \; T(c_1, rdfs{:}\, subClassOf, c_3)$$

$$\forall c_1, c_2 \;\; T(c_1, parentOf, c_2) \Rightarrow \; T(c_2, childOf, c_1)$$

2.8.2 Taxonomic Relations (is-a)

This refers to a relationship between terms that belong to the same category by linking a specific and general term together. For example, between “camel” and “animal”, “orange” and “fruit” and “car” and “vehicle”. The specific term, which is “camel”, “orange” and “car” in this case, is known as hyponym. While the general terms like “animal”, “fruit” and “vehicle” are called the hypernym or superordinate. Taxonomy is also called hierarchical or subordinate relation and taxonomy relation and “is-a”. It is considered as the backbone of the ontology as it is the most common relation found in ontologies. In OWL the notation the built-in relation “rdfs:subClassOf” property links between classes and is a transitive relation (see example above). The notation “rdf:type” property is used to link between individuals and classes “instance-of”. For example, if A is kind of B; A is subordinate to B; A is narrower than B; B is broader than A.

```
<rdfs:label xml:lang="ar">هابيل</rdfs:label>
<rdfs:label xml:lang="en">Abel</rdfs:label>
```

2.8.2.1 Synonym (similar)

This relation refers to relationships between terms that share similar meaning. For example, “desk” and “office”, “announce” and “declare”. In OWL, “owl:sameAs” property is used to link an individual to another individual. While “owl:equivalentClass” is an axiom used to link a class with another class, Relation between two words A is equivalent with B. The following example demonstrates the use of synonym in OWL.

```
<rdf:Description rdf:about="#rabbi_l-ʿālamīna">
  <owl:sameAs rdf:resource="#Allah"/>
</rdf:Description>
```

2.8.2.2 Antonymy (oppositeness)

In contrast to synonym, antonymy refers to a relationship between terms that have opposite meaning of each other or the relation which holds contradictory terms [6][49]. For example, “thin” and “thick”, “high” and “low”, “cold” and “hot”. The built-in notation “owl:disjointWith” in OWL, can be used for representing the disjointness between two entities. It guarantees that an entity cannot be an instance of both disjoint entities. The following example shows the use of antonymy use in food ontology.

```
<owl:Class rdf:ID="Pasta">
  <rdfs:subClassOf rdf:resource="#Flower" />
  <owl:disjointWith rdf:resource="#Fruit" />
</owl:Class>
```

2.8.2.3 Meronymy (Part-Whole)

This refers to a relationship between terms where one of them contains the other. For example, body and hand, car and wheel, computer and processor. Hand, wheel in these examples are called part and body, car are called whole; the wheel is a meronym of the car. It is common in domains like biology, medicine, geographic information systems. For example, a finger is a part of hand and eye is a part of head. Eye is a part of a person, wing is a part of a pigeon.

$$X\ is\ a\ meronym\ of\ Y\ if\ Xs\ are\ parts\ \ of\ Y(s)$$

2.9 Qur’anic Relations

As explained in Section 2.8.1, relations can occur among parts of texts. The text of the Qur’an has its own relations based on its structure and its complexity as can be seen in Figure 2.6. A chapter can be part of Hizb or Juz while Hizb also can be part of Juz. Another relation is between word and segment in which segment is a part of a word. Other relations can be found in the Qur’an as outlined in the following:

- Relations between verse and chapter (meronymy) (a verse is part of chapter)
- Relation between topic and verse (a topic for a verse)
- Relation between title and chapter (a chapter has a title)
- Relation between chapter and Juz (a chapter is part of a Juz)

2.10 Summary

This chapter gave background information on the topics of the thesis. Firstly, it gave an overview of the Qur’an for those who have not heard about it before. Secondly, it focused on ontology and its contents, paying more attention to ontology construction from textual resources. This chapter defined and explained terms and methods used later in this thesis.

Chapter 3

Literature Review and Related Work

This chapter reviews previous work relevant to this thesis in two main parts: existing Qur'anic ontologies and annotations, and ontology learning from textual resources. The first part compares existing Qur'anic ontologies according to a range of appropriate criteria including an analysis of the limitation of previous ontology work for the Qur'an. Following the review of existing ontologies, concept identification approaches with a focus on Arabic language and the Qur'anic text. In addition, hierarchical relations extraction approaches for the Qur'anic domain is also covered. The first part of this chapter, which surveys existing Qur'anic ontologies is based on our published paper [50].

3.1 Qur'anic Ontologies

Recent advances in Text Mining and Natural Language Processing have enabled the development of semantic analysis for religious text and have increased the amount of free online resources. The availability of information is a key factor in knowledge acquisition. Sharing information is an important reason for developing an ontology. This section reports on a survey of recent Qur'an ontology research projects, comparing them in

9 criteria. We conclude that most of the ontologies built for the Qur'an are incomplete and/or focused in a limited specific domain. There is no clear consensus on the semantic annotation format, technology to be used, or how to verify or validate the results.

3.1.1 Comparison Criteria

Table 3.1 in page 45 summarises the comparison of the Qur'an ontologies described in the literature. The comparison focuses on the content of ontologies in the work reviewed. The list of criteria used for comparison is described briefly in the following.

1. **Qur'an text:** The ontology supports/relies on one of the following languages:
    - Original Arabic text (A).
    - English translation (B).
    - Malay translation (C).
    - Bilingual (D).
    - Multilingual (E).

This criterion has been chosen because we noticed that there is a variation of the language used in ontology-based work. For example [17] and [15] ontologies used English translations of the Qur'an, while [16] used a Malay translation. This aspect should not be ignored in research on reusing an ontology as it identifies a challenge in merging different translations of Qur'an ontologies. We also found some support bilingual (Arabic/English) like [51], while others partly support Arabic for some concepts such as [52]. Multilingual is found in [53].

2. **Coverage area:** Topics and word types that are covered by the ontology. For example, an ontology may covers the topic of animals for only nouns. This aspect compares the ontologies on the topic that they have created for.

3. **Coverage proportion:** This criterion identifies if the ontology covers the entire Qur'an or only some parts.

- Entire the Qur’an (A).
- Some parts (B).
- Specific topic (C).

We found only one work that covers all Qur’an chapters, others focused on one or two topics. However, it only covers the personal pronouns.

4. **Underlying format:** There are many formats such as plain text, XML files, and RDF or OWL.
   Format is also an important factor in ontology reuse due to requirements for extra work in extraction of ontology elements from the existing ontologies.

5. **Underlying technology used:** tools used for building and representing the ontology.

6. **Availability:** this criterion identifies if the ontology is free access or not. This is important in reusing an ontology too because we have noticed that there are some resources which are not available for download and reuse.
   - Yes (A)
   - No (B)

7. **Concepts number:** The number of abstract and concrete concepts in the ontology.

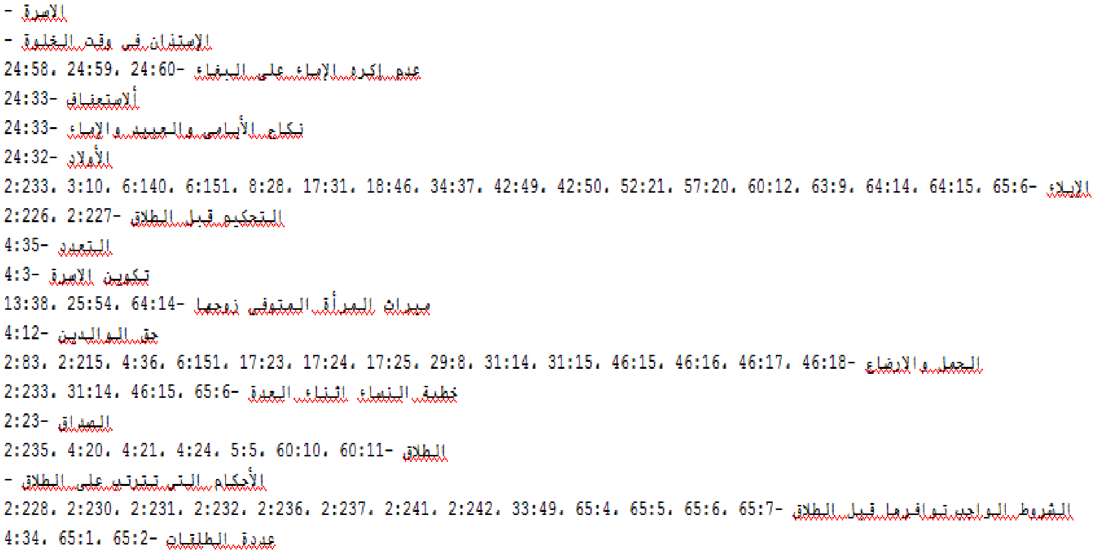
8. **Relations type and number:** The ontology may be have one of the following relations between the concepts:
   - Meronymy (Part-of) (A)
   - Synonyms (similar) (B)
   - Antonymy (opposite) (C)
   - Hyponymy (subordinate) (D)

9. **Verification method used:** The evaluation method used to verify the ontology. Two types of methods have been used to verify ontology-based work on the Qur’an:
   - Domain experts
   - Scholarly sources (Ibn Kathir)

3.1.2 Available Qur’anic Ontologies and Annotations

Many researchers were attracted by modelling Qur’anic knowledge into the so-called ontology. Relevant work on developing Qur’anic ontologies can be classified into two main types; ontology constructed based on mapping available sources, which the ontology elements, in this case, is transformed based on available resources and corpora. The second type is ontology learning from text, in which the basic elements of ontology are learned from the text.

An example of building ontology based on mapping available source can be seen in Qurany [48], which is a web-based search tool. Qurany uses verses topics from available resource called “Mushaf Al Tajweed” [54], these topics were encoded into a dataset and used for the search. The links were made between verses and their topics and another type of relation linked topics and their subtopics. This ontology is available in html format which is not easy to be accessed by other software or downloaded by interested researchers. However, the same information can be encoded from the original source of “Mushaf Al Tajweed” [54] or it can be downloaded from [55], which offers the same list of topic index belong to “Mushaf Al-Tajweed”.



```
- الاسرة
- الاستئذان فى وقت الخلوة
24:58، 24:59، 24:60- عدم إكره الإماء على البغاء
24:33- ألاستعفاف
24:33- نكاح الأيامى والعبيد والإماء
24:32- الأولاد
2:233، 3:10، 6:140، 6:151، 8:28، 17:31، 18:46، 34:37، 42:49، 42:50، 52:21، 57:20، 60:12، 63:9، 64:14، 64:15، 65:6- الإيلاء
2:226، 2:227- التحكيم قبل الطلاق
4:35- التعدد
4:3- تكوين الاسرة
13:38، 25:54، 64:14- ميراث المرأة المتوفى زوجها
4:12- حق الوالدين
2:83، 2:215، 4:36، 6:151، 17:23، 17:24، 17:25، 29:8، 31:14، 31:15، 46:15، 46:16، 46:17، 46:18- الحمل والارضاع
2:233، 31:14، 46:15، 65:6- خطبة النساء اثناء العدة
2:23- الصداق
2:235، 4:20، 4:21، 4:24، 5:5، 60:10، 60:11- الطلاق
- الأحكام التى تترتب على الطلاق
2:228، 2:230، 2:231، 2:232، 2:236، 2:237، 2:241، 2:242، 33:49، 65:4، 65:5، 65:6، 65:7- الشروط الواجب توافرها قبل الطلاق
4:34، 65:1، 65:2- عدة الطلقات
```

Figure 3.1 The extracted topic index in text file

Figure 3.1 shows the extracted text file from [55] website. The downloaded text file contains the topics of the Qur’an as a nested list and each element of the list denotes a topic. Each element is followed by their related verses and abstract topic are followed by subtopics instead of verses.

Another research project on a prototype of a framework called SemQ is presented in [56]. SemQ identifies opposition relationships between Qur'anic concepts. The idea is SemQ receives a verse as input and produces a list of words that are opposed to each other with the degree of the opposition. The coverage is in the domain of "Women" in the Qur'an. Ontology development makes use of the Buckwalter morphology POS annotation and focuses on nouns and verbs that are related to the semantic field of Time. This paper used OWL and UPON technologies in order to represent the concepts and relations. The ontology consists of seven abstract concepts and eleven concrete concepts. This ontology is sharable and can be downloaded. This study was limited to word level which includes only nouns and verbs of the "Women" domain. However, there are no evaluable results provided by the authors or any validation attempts for their proposed framework.

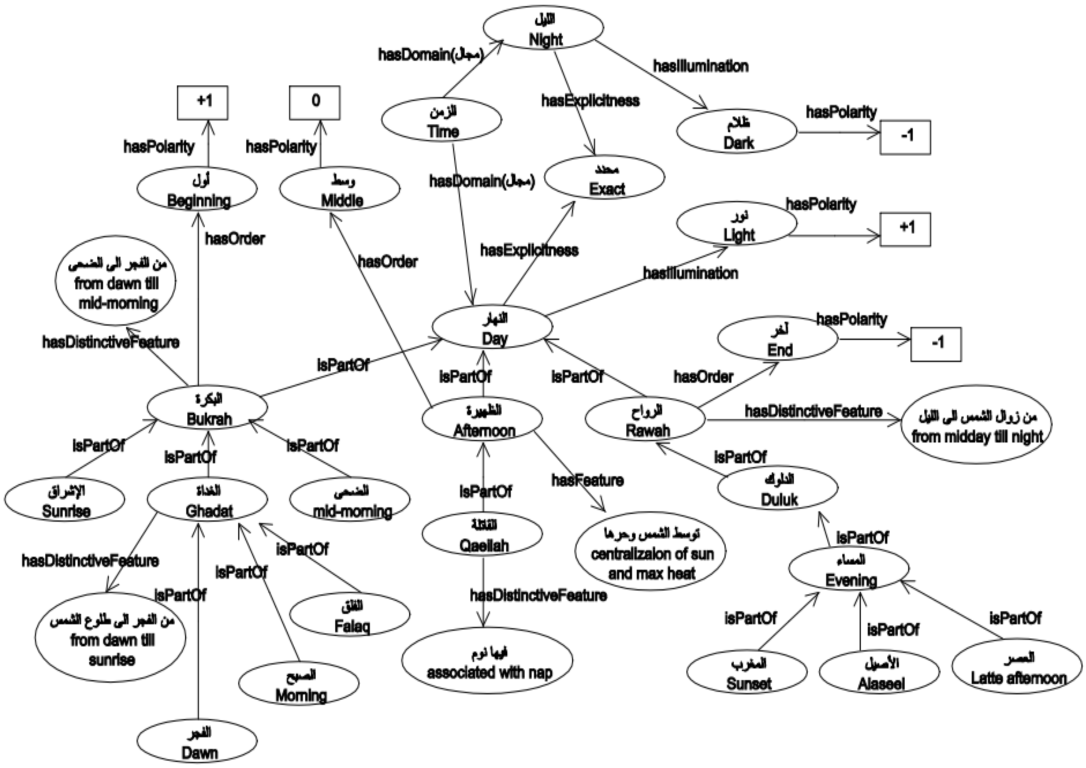


Figure 3.2 sample of SemQ ontology individuals proposed in [56]

A theme-based approach is proposed to represent and classify the knowledge of the Qur'an using an ontology in [16]. Their ontology was developed according to themes described in Syammil Al-Qur'an Miracle the Reference, and using protégé-OWL and Malay language as medium of concepts, and was validated by the domain experts. It only covers two themes: "Iman" which means faith and "Akhlaq" which means deed. This was an Ontology-based approach to represent and classify Qur'anic concepts according to specific semantic fields. The structure of the ontology was verified by Qur'an domain experts. The ontology was developed using Protégé-OWL and using Malay Language as the medium language. The authors proposed a representation approach whcih differs from traditional representation which consist of Juz, Chapter and Verse. There is no explanation of what language was used for this ontology and what source the concepts were based on. They implemented the ontology using protégé. There are no details of results or validation of the ontology, although the paper states that the process of creating the ontology was reviewed by seven Qur'an domain experts.

A simple ontology for the Qur'an that includes the animals that are mentioned in the Qur'an in order to provide Qur'anic semantic search was developed by [17]. The ontology was built using protégé editor, and SPARQL query language was used to retrieve the answers to a query. The English translation of the Qur'an by Pickthall is used in this ontology. The ontology provides 167 direct or indirect references to animals in the Qur'an obtained based on information mentioned in a book entitled "Hewanat-E-Qurani". The relationship type is a taxonomy relation. The paper concludes that the existing Arabic WordNet does not help for retrieving this type of document information.

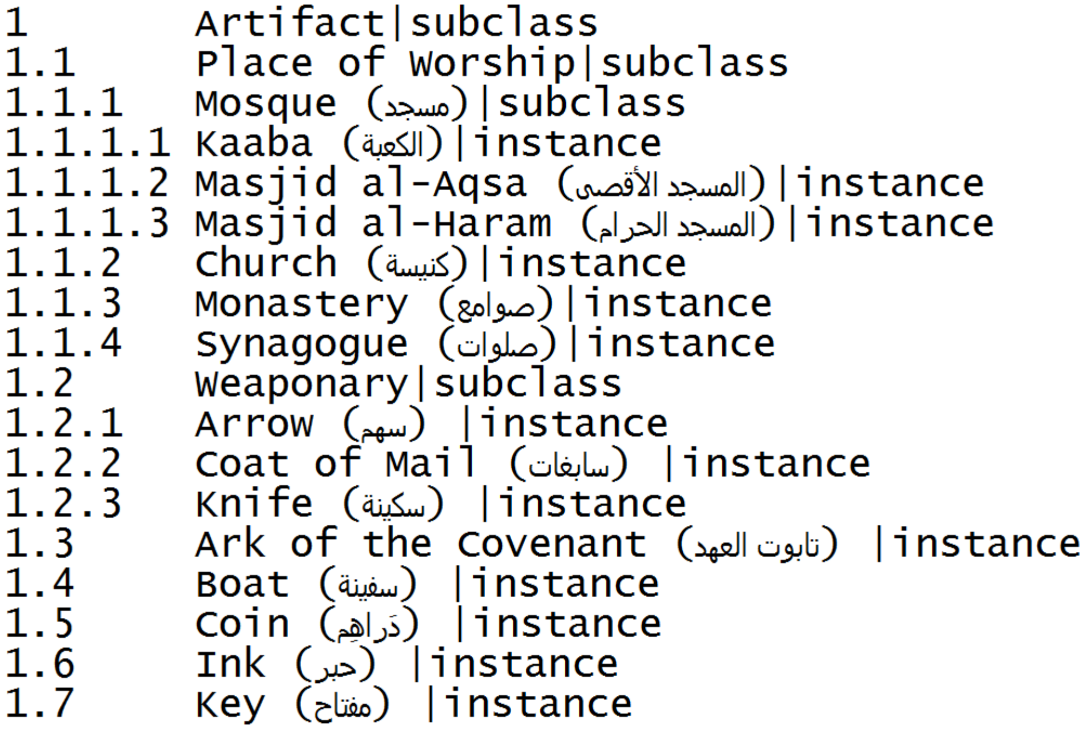
The work reported in [57] proposed a model for defining the important Qur'anic concepts by knowledge representation and presented the relationships between them using Description Logic (Predicate logic). They reused the Qur'anic Arabic Corpus ontology by (Dukes 2013). This ongoing research attempts to reuse and improve an existing ontology developed in Leeds by adding more relations. Protégé is used in ontology construction. A top-down ontology development process was followed. It has 15 abstract concepts.

Another work presented in [18] proposed ontology-based semantic search for the Qur'an using protégée editor and Manchester OWL query language. The ontology was built by reusing the existing Qur'anic Arabic Corpus ontology developed by (Dukes 2013), and adding more than 650 relationships depending on the Qur'an, Hadith, and Islamic websites. This ontology was constructed manually.

Similarly, the work reported by [19] proposed a semantic search for the Qur'an based on Cross Language Information Retrieval (CLIR). They created a bilingual ontology for the Qur'an composed of concepts based on existing Qur'anic Arabic Corpus ontology by (Dukes 2013), and found 5695 documents belonging to a main concept, where 541 documents are not assigned to any concepts in an English translation. In Malay, there are 5999 documents assigned to main concepts, where 237 documents do not belong to any concept.

In [20], the authors did experiments on retrieving verses of the Qur'an using a semantic query approach exploiting Cross Language Information Retrieval (CLIR). The Arabic text and two translations: Malay and English have been used in this experiment in order to retrieve the ontological information about the Quran. Unfortunately, the documents of this experiment was not available online.

Another similar work reported by [21] in which his PhD thesis defines 300 concepts for the Qur'an, and extracts the interrelationships using Predicate logic. The number of relations are 350 and the type of relation between concepts is IS-A. The ontology is also based on the Tafsir by Ibn Kathir. This ontology is not available in text format, but available in html format.



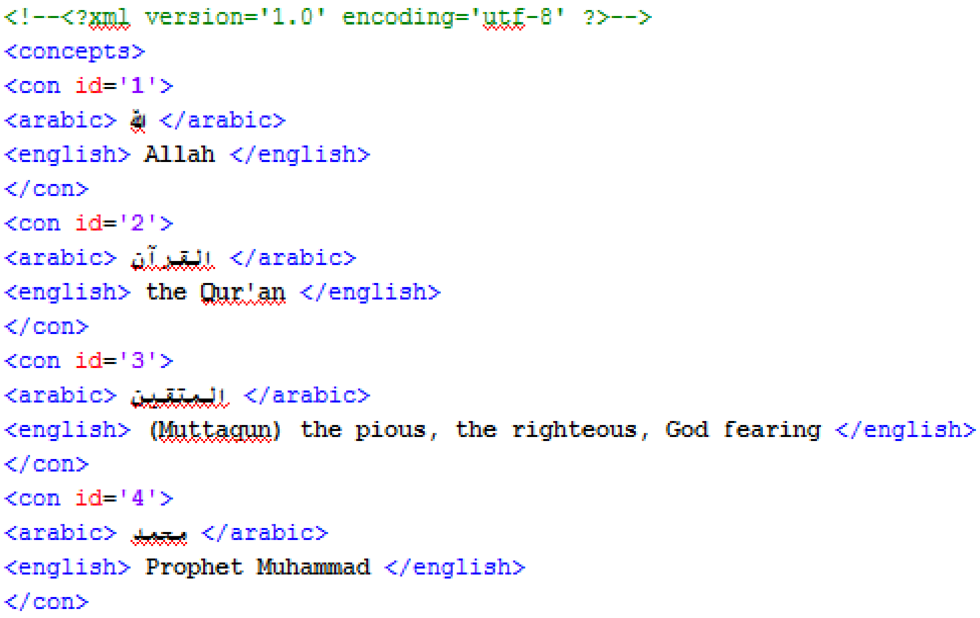
```
1         Artifact|subclass
1.1       Place of Worship|subclass
1.1.1     Mosque (مسجد)|subclass
1.1.1.1 Kaaba (الكعبة)|instance
1.1.1.2 Masjid al-Aqsa (المسجد الأقصى)|instance
1.1.1.3 Masjid al-Haram (المسجد الحرام)|instance
1.1.2     Church (كنيسة)|instance
1.1.3     Monastery (صوامع)|instance
1.1.4     Synagogue (صلوات)|instance
1.2       Weaponary|subclass
1.2.1     Arrow (سهم) |instance
1.2.2     Coat of Mail (سابغات) |instance
1.2.3     Knife (سكينة) |instance
1.3       Ark of the Covenant (تابوت العهد) |instance
1.4       Boat (سفينة) |instance
1.5       Coin (دَراهِم) |instance
1.6       Ink (حبر) |instance
1.7       Key (مفتاح) |instance
```

Figure 3.3 QAC ontology

Figure 3.3 presents a sample of the information extracted from QAC. Each line in the figure represents a concept in the QAC with the number of the concept in the left column and the Arabic and English translation of the concept on the second column. The subsection number “1.1” means the corresponding concept is a sub concept of the previous concept. For example the sub concepts “1.1”, “1.3”, “1.4”, “1.5”, “1.6” and “1.7” are sub concepts of the concept “1”.

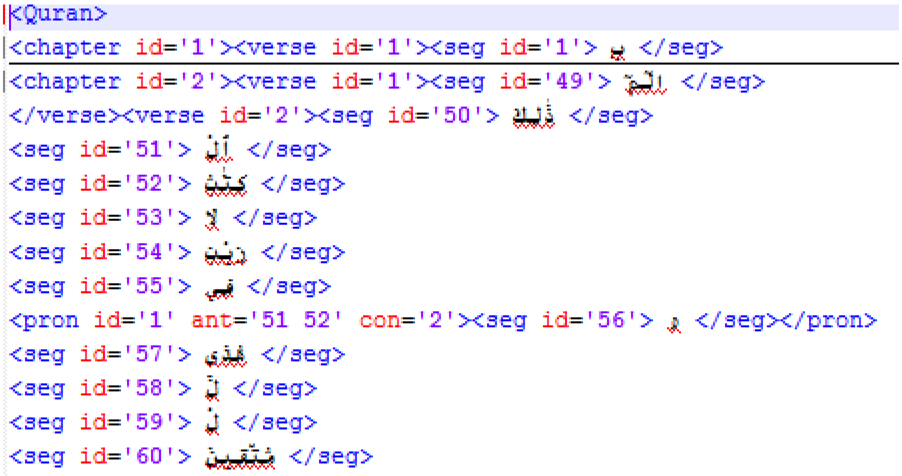
Sharaf & Atwell in [58], have created a dataset called QurSim which consists of 7600 pairs of related verses for evaluating the relatedness of short texts. The similarity between verses were based on the comments of the Ibn Kathir’s Tafsir, which is one of the most widely cited Tafsir. This dataset is available in xml and MySQL format for download.

Sharaf & Atwell [41], [22] developed an ontology that encompassed the entire Qur’an in terms of personal pronoun tagging called QurAna, whereby each pronoun is linked to its syntactic antecedent or previous reference, and its concept in an ontology of pronoun referrents. The dataset comprises about 24,000 personal pronouns in the Arabic text, each linked to its antecedent and its concept in the ontology. This can be used in ontology extraction in an ontology learning system using anaphora analysis to extract the concepts and relationships. Over 1,000 antecedents which are also domain-specific terms about the text, were arranged in a dataset called QurAna. . This dataset is available in xml and MySQL database formats and can be very useful for



```
<!--<?xml version='1.0' encoding='utf-8' ?>-->
<concepts>
<con id='1'>
<arabic> الله </arabic>
<english> Allah </english>
</con>
<con id='2'>
<arabic> القرآن </arabic>
<english> the Qur'an </english>
</con>
<con id='3'>
<arabic> المتقين </arabic>
<english> (Muttaqun) the pious, the righteous, God fearing </english>
</con>
<con id='4'>
<arabic> محمد </arabic>
<english> Prophet Muhammad </english>
</con>
```

Figure 3.4 The referents list of QurAna



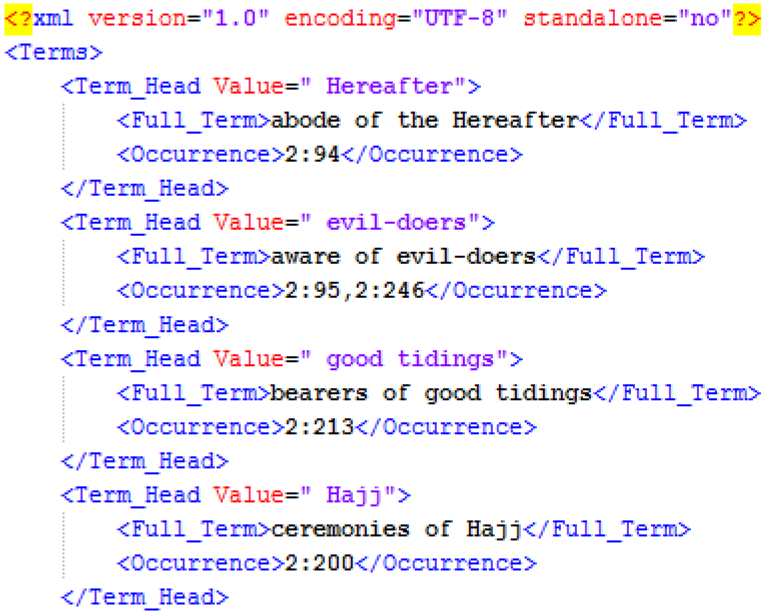
```
<Quran>
<chapter id='1'><verse id='1'><seg id='1'> بِ </seg>
<chapter id='2'><verse id='1'><seg id='49'> الٓمٓ </seg>
</verse><verse id='2'><seg id='50'> ذَٰلِكَ </seg>
<seg id='51'> ٱلْ </seg>
<seg id='52'> كِتَٰبُ </seg>
<seg id='53'> لَا </seg>
<seg id='54'> رَيْبَ </seg>
<seg id='55'> فِي </seg>
<pron id='1' ant='51 52' con='2'><seg id='56'> هِ </seg></pron>
<seg id='57'> هُدًى </seg>
<seg id='58'> لِّ </seg>
<seg id='59'> لْ </seg>
<seg id='60'> مُتَّقِينَ </seg>
```

Figure 3.5 The segment and their antecedent information

An automatic knowledge extraction method based on rules and natural language patterns is described in [59]. Their methods rely on the English translation of the Qur'an and have identified a new pattern language named Qpattern which is suitable for extraction of taxonomy part-of relations. This research also identified that it is difficult to extract information from text that includes co-reference like the Qur'an.

The aim of this work was to look at the range of existing studies on Qur'an ontology available currently and identify the limitations of these studies as well as potential future work. Some semantic annotations have been done for the entire Qur'an, but for a specific type of word and domain, such as [56], an ontology for verbs in the domain of women, or the ontology of [60] for nouns in the domain of time. There is one non-domain-specific ontology for the entire Qur'an but it is only for pronouns [22]. Most ontologies have relations using Part-Of or synonyms, but one work includes opposition relations, [56]. Most ontologies built for the Qur'an are incomplete and focused in a specific domain. There is no clear consensus on the semantic annotation format, technology to be used, or how to verify or validate the results.

[61] produced a dataset that contains 693 vocabularies extracted from the second chapter of the Qur'an, known as the Vocabulary of Qur'anic Concepts (VOQ). The extracted terms provided in sql and xml format. Authors used six different English translations of the Qur'an and applied a domain-independent tool called Termine from [62] to extract the concepts. A sample of the VOQ can be seen in Figure 3.6.



```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<Terms>
    <Term_Head Value=" Hereafter">
        <Full_Term>abode of the Hereafter</Full_Term>
        <Occurrence>2:94</Occurrence>
    </Term_Head>
    <Term_Head Value=" evil-doers">
        <Full_Term>aware of evil-doers</Full_Term>
        <Occurrence>2:95,2:246</Occurrence>
    </Term_Head>
    <Term_Head Value=" good tidings">
        <Full_Term>bearers of good tidings</Full_Term>
        <Occurrence>2:213</Occurrence>
    </Term_Head>
    <Term_Head Value=" Hajj">
        <Full_Term>ceremonies of Hajj</Full_Term>
        <Occurrence>2:200</Occurrence>
    </Term_Head>
```

Figure 3.6 A snapshot of Vocabulary of Qur’anic Concepts extracted by [61]

Another ontology which is encoded using a formal ontology language (OWL), is reported in [63]. This ontology was mapped from different Qur’anic datasets. These datasets were generated manually based on what has mentioned in commentary books, so the set of concepts which this type of ontologies may include are some equivalent concepts rather than the actual words used in the original texts. Therefore, most of these concepts have no any kind of relations. Most of provided relations are between verses and topics, topics and subtopics and chapters and verses, which represent the structure of the book rather than relations between the content of the Qur’an.

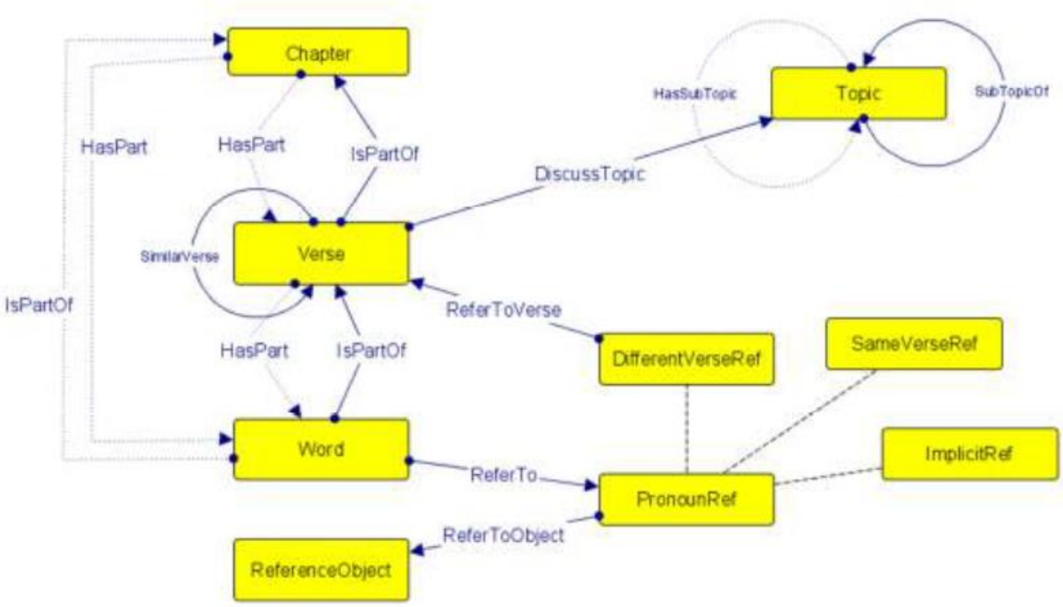


Figure 3.7 Sechme used for Qur'an ontology by [63]

Another similar work of mapping is the Semantic Qur'an, which was manually built using protégé by the authors of [53], who mapped Qur'anic data from two different sources: Tanzil project and Qur'anic Arabic Corpus into RDF representation. Then a tool called LIMES was used to link the terms [64].

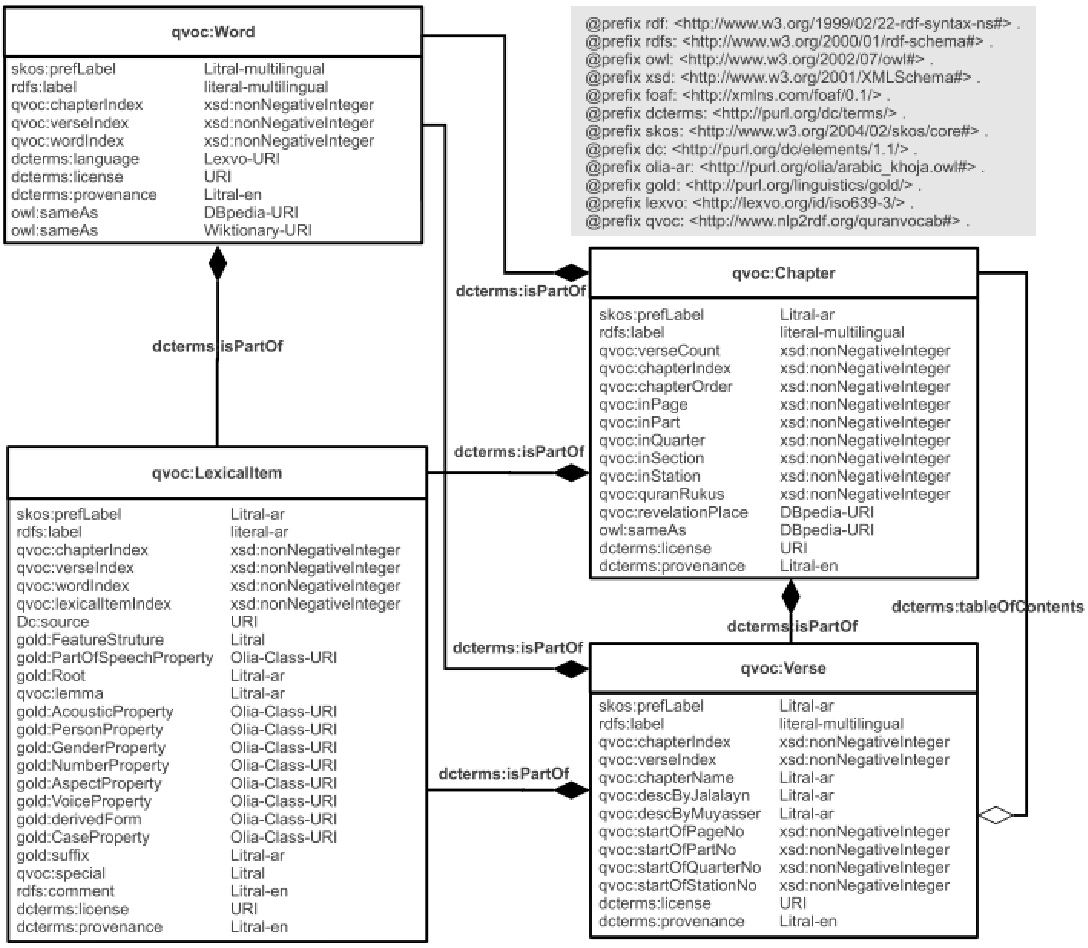

qvoc:Word
skos:prefLabel Litral-multilingual
rdfs:label literal-multilingual
qvoc:chapterIndex xsd:nonNegativeInteger
qvoc:verseIndex xsd:nonNegativeInteger
qvoc:wordIndex xsd:nonNegativeInteger
dcterms:language Lexvo-URI
dcterms:license URI
dcterms:provenance Litral-en
owl:sameAs DBpedia-URI
owl:sameAs Wiktionary-URI
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix skos: <http://www.w3.org/2004/02/skos/core#> .
@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix olia-ar: <http://purl.org/olia/arabic_khoja.owl#> .
@prefix gold: <http://purl.org/linguistics/gold/> .
@prefix lexvo: <http://lexvo.org/id/iso639-3/> .
@prefix qvoc: <http://www.nlp2rdf.org/quranvocab#> .
dcterms:isPartOf
dcterms:isPartOf
qvoc:Chapter
skos:prefLabel Litral-ar
rdfs:label literal-multilingual
qvoc:verseCount xsd:nonNegativeInteger
qvoc:chapterIndex xsd:nonNegativeInteger
qvoc:chapterOrder xsd:nonNegativeInteger
qvoc:inPage xsd:nonNegativeInteger
qvoc:inPart xsd:nonNegativeInteger
qvoc:inQuarter xsd:nonNegativeInteger
qvoc:inSection xsd:nonNegativeInteger
qvoc:inStation xsd:nonNegativeInteger
qvoc:quranRukus xsd:nonNegativeInteger
qvoc:revelationPlace DBpedia-URI
owl:sameAs DBpedia-URI
dcterms:license URI
dcterms:provenance Litral-en
dcterms:isPartOf
qvoc:LexicalItem
skos:prefLabel Litral-ar
rdfs:label literal-ar
qvoc:chapterIndex xsd:nonNegativeInteger
qvoc:verseIndex xsd:nonNegativeInteger
qvoc:wordIndex xsd:nonNegativeInteger
qvoc:lexicalItemIndex xsd:nonNegativeInteger
Dc:source URI
gold:FeatureStruture Litral
gold:PartOfSpeechProperty Olia-Class-URI
gold:Root Litral-ar
qvoc:lemma Litral-ar
gold:AcousticProperty Olia-Class-URI
gold:PersonProperty Olia-Class-URI
gold:GenderProperty Olia-Class-URI
gold:NumberProperty Olia-Class-URI
gold:AspectProperty Olia-Class-URI
gold:VoiceProperty Olia-Class-URI
gold:derivedForm Olia-Class-URI
gold:CaseProperty Olia-Class-URI
gold:suffix Litral-ar
qvoc:special Litral
rdfs:comment Litral-en
dcterms:license URI
dcterms:provenance Litral-en
dcterms:tableOfContents
dcterms:isPartOf
dcterms:isPartOf
qvoc:Verse
skos:prefLabel Litral-ar
rdfs:label literal-multilingual
qvoc:chapterIndex xsd:nonNegativeInteger
qvoc:verseIndex xsd:nonNegativeInteger
qvoc:chapterName Litral-ar
qvoc:descByJalalayn Litral-ar
qvoc:descByMuyasser Litral-ar
qvoc:startOfPageNo xsd:nonNegativeInteger
qvoc:startOfPartNo xsd:nonNegativeInteger
qvoc:startOfQuarterNo xsd:nonNegativeInteger
qvoc:startOfStationNo xsd:nonNegativeInteger
dcterms:license URI
dcterms:provenance Litral-en
dcterms:isPartOf

Figure 3.8 Scheme used for smanticQur'an by [64]

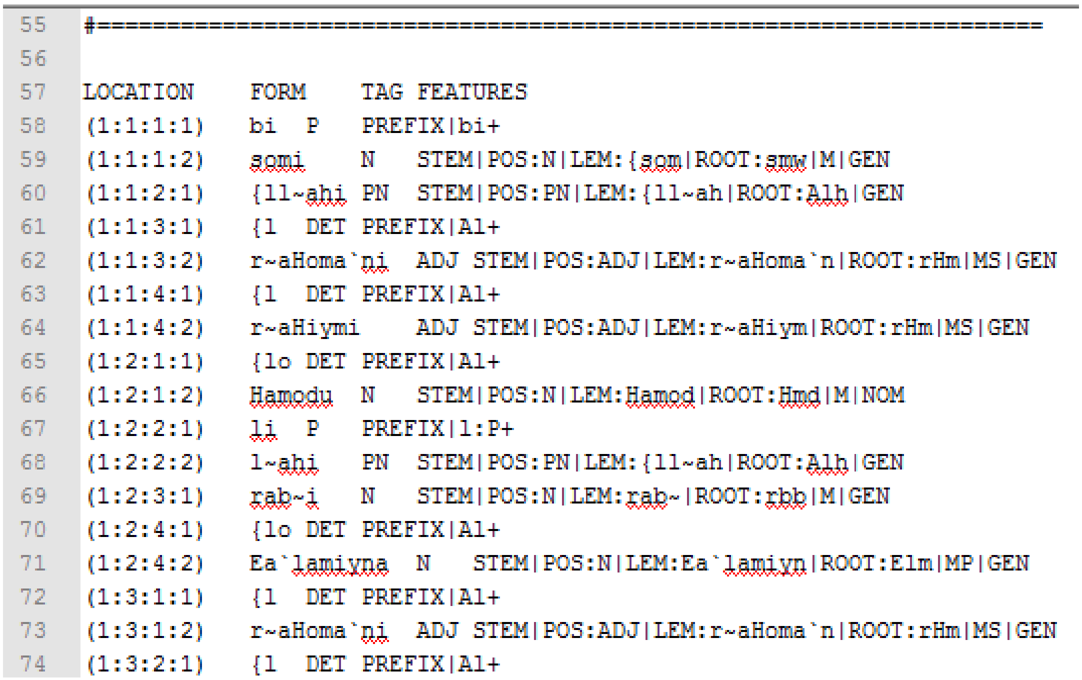
Ontology-based models of computational semantics are being widely adopted in various fields such as Knowledge Management, Information Extraction, and the Semantic Web. Religious Studies researchers are also starting to exploit the ontology for improving the capture of knowledge from religious texts such as the Qur'an and Hadith. A definition of ontology in Artificial Intelligence is "the specification of conceptualizations, used to help programs and humans share knowledge.”. Ontology development is generally described as an iterative process, and the development process never completes [17]. Therefore, many researchers start with a focus on one or two semantic fields of their Qur'an ontology.

There are different annotations and ontologies already available for the Qur'an online and most of them are free. However, they differ in the format that they provide for End-Users, and in the technologies that they use to construct and implement the ontology. This variety of formats used to store the annotated data of the Qur'an leads to a gap between computer scientists, who make tools to provide analysis, and End-Users who are interested in the specific domain. Not all End-Users or linguistics researchers are technically able or willing to make their own converter. Therefore, the need to design a standard format and provide available analyses for the Qur'an in a standard format is becoming essential to facilitate End-User work and make them focus on the analysis instead of doing extra data-reformatting work. Moreover, this would increase the process of development in Qur'an analysis. This paper does not try to solve these challenges, instead of that it tries to survey the Qur'an ontology research projects that have been done recently, comparing them in terms of 9 criteria.

A simple methodology for automatic extraction of a concept based on the Qur'an in order to build an ontology was proposed by [65]. This paper used a method based on extracting the verses which contain a word of prayer in it as well as the previous and next verse. This method relies on a format of one English translation of the Qur'an that included some aspects such as Uppercase Letter. An uppercase letter is used to identify the concepts such as the Book. Another feature called compound noun is used to identify relationships such as hyponym and "Part-OF" between the concepts. A copula is used to identify the syntactic relationship between subject and adjective or noun. The ontology is based on the information obtained from domain experts. The development process is adopted from [66]. However, the authors have focused on the subject of Prayer or "Salat" particularly in daily prayer, thus this ontology does not cover all subjects in the Qur'an. In addition, there is no mention about underlying format or ontology technologies used in this paper. In addition, some verses have been extracted their concepts without relations.

The author of [65], has continued their work to develop a framework for automated generation of Islamic knowledge concrete concepts that exist in the holy Qur'an as presented in [15]. The framework takes into account some situations form the sciences of the Qur'an, such as the cause of revelation (Asbab Al Nuzul), and verses overridden by related verses that were revealed later (Nasikh Mansukh). The methodology of ontology development was also adopted from [66], and the method to obtain the concepts is applying a grammar and extraction rules to the English translation of the Qur’an. The 374 extracted instances only cover verses that have the keyword salah or pray and this does not cover the entire Qur’an. These instances were mapped to six abstract concepts. This paper differs from the previous in synonym relations.

Another methods for designing an ontology based on translated texts of the Qur’an is proposed by [67]. Information used in developing the ontology was collected by the domain experts. Their ontology also only covers the subject of “Salat” (pray).



```
#====================================================================

LOCATION     FORM    TAG FEATURES
(1:1:1:1)    bi   P    PREFIX|bi+
(1:1:1:2)    somi     N    STEM|POS:N|LEM:{som|ROOT:smw|M|GEN
(1:1:2:1)    {ll~ahi PN   STEM|POS:PN|LEM:{ll~ah|ROOT:Alh|GEN
(1:1:3:1)    {l   DET PREFIX|Al+
(1:1:3:2)    r~aHoma`ni  ADJ STEM|POS:ADJ|LEM:r~aHoma`n|ROOT:rHm|MS|GEN
(1:1:4:1)    {l   DET PREFIX|Al+
(1:1:4:2)    r~aHiymi     ADJ STEM|POS:ADJ|LEM:r~aHiym|ROOT:rHm|MS|GEN
(1:2:1:1)    {lo DET PREFIX|Al+
(1:2:1:2)    Hamodu   N    STEM|POS:N|LEM:Hamod|ROOT:Hmd|M|NOM
(1:2:2:1)    li   P    PREFIX|l:P+
(1:2:2:2)    l~ahi    PN   STEM|POS:PN|LEM:{ll~ah|ROOT:Alh|GEN
(1:2:3:1)    rab~i    N    STEM|POS:N|LEM:rab~|ROOT:rbb|M|GEN
(1:2:4:1)    {lo DET PREFIX|Al+
(1:2:4:2)    Ea`lamiyna   N    STEM|POS:N|LEM:Ea`lamiyn|ROOT:Elm|MP|GEN
(1:3:1:1)    {l   DET PREFIX|Al+
(1:3:1:2)    r~aHoma`ni  ADJ STEM|POS:ADJ|LEM:r~aHoma`n|ROOT:rHm|MS|GEN
(1:3:2:1)    {l   DET PREFIX|Al+
```

Figure 3.9 The morphological annotation from QAC

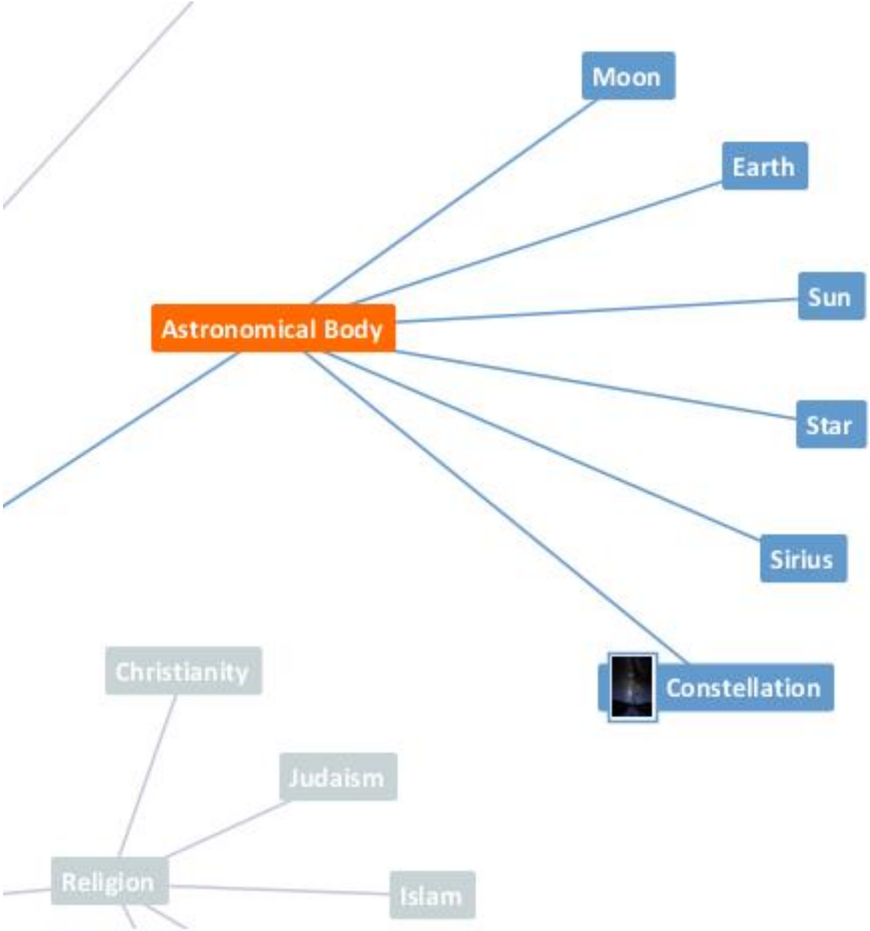


Figure 3.10 Partial concept map from QAC for "Astronomical Body" are highlighted.

Reference	1	2	3	4	5	6	7	8	9
[65]	B	“Pray”	C	N/A	N/A	N/A	N/A	A, Part-Of	Domain experts
[15]	B	“Pray”	C	N/A	N/A	N/A	374 instances and 6 abstract	B, synonyms	Domain experts
[56]	A	“Women”, Nouns and Verb	C	OWL	UPON	A	11 concrete and 7 abstract	C, opposition	N/A
[60]	A	“Time noun”	C	OWL	UPON	A	11 concrete and 7 abstract	D, hyponymy	N/A
[16]	C	“faith and deed”	C	OWL	protégé	N/A	N/A	N/A	Domain experts
[17]	B	“animals”	C		Protégé, SPARQL	N/A	N/A	A, Part-od	N/A
[57]	N/A	”salat, zakat, sin, reward”	C	OWL	Protégé	N/A	15 abstract	N/A	N/A
[18]	N/A	N/A	N/A	Manchester OWL	N/A	N/A	N/A	N/A	Manually constructed
[19]	B, C	N/A	C	N/A	N/A	N/A	N/A	N/A	N/A

[20]	A, B, C	N/A	C	N/A	N/A	N/A	N/A	N/A	N/A
[21]	A, B	N/A	N/A	Text files	N/A	B	300	Part-of	Ibn Kathir
[22]	A	pronouns	A	XML	N/A	A	N/A	N/A	Ibn Kathir
[61]	B	N/A	B	Sql and xml files			578		

Table 3.1 The summary of ontology features from our early survey in [50]

3.2 Ontology Learning From Textual Resources

Recently, several approaches have been proposed for ontology learning from textual resources. However, The Arabic language does not benefit from the same level of development on the English language.

Most of ontology learning approaches are focus on two steps of learning: (1) learning the concept. (2) learning the relationships. This section is followed these approaches as it is divided into two subsections; concept identification and relationship extraction.

3.2.1 Concept Identification

An initial step to build a domain ontology is to find the important concepts of that domain [25], [68]. This step is based on the extracted domain-specific terms. Most of concept extraction methods execute this in two steps; generate the candidates, then find the important candidates. However, generating candidates is not easy, as most of domain-specific terms are composed of multi-word terms.

Most existing approaches used for term extraction to date can be divided into three categories: (1) **linguistic** approaches, (2) **statistical** approaches, and (3) **hybrid** approaches. **Linguistic** approaches exploit NLP techniques, such as tokenisation, morphological analysis, POS tagging, stemming and parsing, for detecting terms from a given text. This method is usually dependent on the selected domain and would not work perfectly in other domains because of its language dependency. For example, linguistic-based methods for extracting medical terms search for some medical characteristics in the text itself, such as abbreviations and doctors' instructions, while other domains do not have the same characteristics. Based on linguistic aspects for example, [69] who based on specific structure found in Wikipedia.

Statistical approaches overcome language dependency because they rely on independent measures for assessing the importance of extracted candidates; measures such as frequencies, likelihood, term frequency-inverse document frequency (TF-IDF) and mutual information, can be calculated for any domain. However, some statistical methods are incapable of addressing low-frequency terms. In fact, the majority of the words in most corpora have low frequencies [37], especially MWTs occurring only once or twice. This means that the MWTs are excluded by statistical approaches. For Arabic language statistical methods should be combined with linguistic methods for Arabic language [37], [70].

Hybrid approaches combine different methods from linguistics and statistics for detecting terms in the text. Firstly, applying the linguistic method to generate term candidates. After that, statistical measures are used for filtering out invalid candidates. Hybrid method tends to give better results for extracting terms from Arabic language [71].

3.2.1.1 Generate Candidate Terms & Relevance Measurement

There are two main approaches to find possible candidate terms from textual resources. The first approach, which used in [72], is to extract single-word terms, then multi-word terms are formed based on statistical measures (i.e. Mutual Information or Co-occurrences) which find collocation occurs among single-word terms. The generated MWTs may not occur in the text and only combined based on the statistical information. Therefore, this approach may miss many important multi-word terms.

The second approach, which adopted by Text-To-Onto in [73] and in [74] , starts by applying a set of predefined linguistic filters (POS tag-based rules) for extracting single-word and multi-word terms. Then apply statistical measurement (i.e. TF-IDF) to find the most important ones. As [68] reported that TF-IDF was not designed for concept extraction, and an important terms may not be selected because its IDF values tends to be 0 [26].

A successor framework for ontology learning from textual resources called Text2Onto by [75], apply same technique. The authors assessed the relevance of terms using Relative Term Frequency (RTF), Term Frequency Inverted Document (TF-IDF), Entropy and C-value/NC-value. However, as reported by [26], RTF and entropy rely only on frequency information which may increase the non-relevant single-word terms. In order to overcome the problem of the TF-IDF, CRCTOL in [68] extract MWTs and single-word are added if they are related to MWTs.

3.2.1.1.1 Related Work for Arabic

Some research has been performed for extracting terms on the Arabic language. We found only three papers on Qur'anic term extraction from Arabic; conducted by [76], [7] and [77]. Therefore, further research is required in this area. Previous studies that examined the extraction of terms from the Arabic text of the Qur'an proposed hybrid methods that are based on syntactic patterns and TF-IDF for extracting single-terms [7], [77].

Zaidi et al [76] extracted simple terms and collocations from the Arabic text of the Qur'an for the purpose of ontology construction. The simple terms, which are composed of single word, extracted using TF-IDF measure. TF-IDF assigned to simple-terms as a weight, then they sorted in descending order and a number of terms that located above the threshold were extracted. The paper did not mention what was that number. For measuring the performance of their approach of extraction, they use a list from [78] as a reference. The results contained some noise due to some words were not segmented accurately such as words has preposition as a prefix. Collocations, which are only composed of two words, were extracted using collocations technique based on a predefined set of two POS tags (Noun-Adjective, Noun-Noun, and Verb-Noun) and implemented using JAPE.

[7] used methods that rely on linguistics and statistical language modelling to generate ontology elements. 12 chapters include prophetic stories were selected for their experiments. The concept extraction was based on KP-Miner, which, in turn, uses TF-IDF, which requires some specifications that are not available for the Qur'an, such as the size of the corpus. No results about the extraction performance were mentioned because the focus was to extract the conceptual relations.

[77] aimed to extract the topic of a single-word from the chapter no 12 of the Qur'an using Arabic text. He applied a package called tm from R language which implements topic modelling from Latent Dirichlet

Allocation (LDA). Basically, the author built a document term matrix where its rows are a set of single-word terms of the chapter no 12 and its columns are the chapters. Each row has been assigned with the probability of the corresponding single-word term under Qur'an chapters. This method suffers from data-sparsity since many words have been mentioned only once and only a few terms have been mentioned in all chapters such as "Allah".

Another related work for Arabic language in Boulaknadel & Daille, (2008) investigated different statistical measures and linguistic techniques for multi-word term extraction for Arabic technical text from environment domain. The methodology started with defining the linguistic specification of MWTs for Arabic language. Then they select MWT based on N1 ADJ| N1 P? N2.After that, they evaluated several statistical measures. Such as Log-likelihood ratio (LLR), FLR, MI3 and t-scores. The extracted terms were compared to a list of known Arabic terms from the same domain. Each extracted term was given scores for the selected statistical measures. Then ranked these terms and take 100 and compare it to the reference list. Only precision was computed where recall was not. FLR measure was the highest with 60%. The authors concluded that LLR achieved the best performance.

[79] developed a general purpose bilingual (English/Arabic) keyphrases extraction system, called KP-Miner, for key-phrases extraction from both Arabic and English documents. Arabic documents were selected from Agriculture domain. The selection of candidates based on a set of rules; (1) that phrases have no punctuation marks or stopwords in the middle. (2) A phrase has to have appeared at least n times. The least allowable seen frequency (lasf) factor. It is a filter of threshold and it was applied to remove all candidates that are less than the threshold (lasf). (4) CutOff, which is a filter to remove all candidates that their first appearance in the text is after a certain position (CutOff). For weighting candidates, KP-Miner combined TF-IDF with a boosting factor that makes the balance between low frequency of compound words and high frequency of single words in a corpus. KP-Miner does not need text to be tagged with POS, does not require training data, but it requires an understanding and some experimental work for the targeted text in order to set some configurations such as "lasf" and "CutOff". The lasf factor depends on the

language and the text size, so it should be set based on various situations, which makes it more difficult than adapting other methods from different domain.

3.2.1.1.2 Related Work for Qur'anic Text

The majority of term extraction research work conducted on the Qur'an have used translated text instead of the original text, such as [80]; however, extraction of terms from the original Arabic text may help in retrieving more relevant terms than an extraction from a translation. This is because some Islamic terms have no equivalents in other languages [14], [81]. Another reason is that some translations do not consider the meaning of the terms. For example in "whoever does not judge by what has revealed Allah". We can only see one term which is "Allah" in the translation of that term.

Another work related to the validation process of our method is (Dukes 2012, 2013) which described the Quranic Arabic Corpus (QAC) project, an online Qur'an that was annotated at several levels, which included an ontology that defines 300 concepts in the Qur'an, and captures interrelationships using predicate logic. The number of relationships is 350, and the type of relationship between concepts is "is-a". The ontology is based on the Tafsir or Qur'an commentary textbook by Ibn Kathir. The QAC also contains other analyses of the Qur'an text, such as POS, morphological analysis and dependency parse structure analysis.

[41] developed an ontology that encompassed the entire Qur'an in terms of pronoun tagging called QurAna, whereby each pronoun is linked to its syntactic antecedent or previous reference, and its concept in an ontology of pronoun referrents. The dataset comprises about 24,000 personal pronouns in the Arabic text, each linked to its antecedent and its concept in the ontology. This can be used in ontology extraction in an ontology learning system using anaphora analysis to extract the concepts and relationships. Over 1,000 antecedents which are also domain-specific terms about the text, were arranged in a dataset called QurAna.

[61] produced a dataset that contains concepts from the second chapter of the Qur'an, known as the Vocabulary of Qur'anic Concepts.

They used six different English translations of the Qur'an and applied a domain-independent tool called Termine from Frantzi & Ananiadou (1999) to extract the concepts. However, Termine was designed for the extraction of multi-word terms, while the Qur'an has numerous single-word concepts (e.g., Allah and Muhammad). Therefore, application of such a method may exclude some important concepts.

Another related work in [7] used methods that rely on linguistics and statistical language modelling to generate ontology elements. Instead of generating the terms this work focus on collocations, only single-word terms was extracted then, collocations generated based on a predefined set of pos rules. The concept extraction was based on KP-Miner, which, in turn, was based on TF-IDF, which requires some specifications that are not available for the Qur'an, such as the size of the corpus. Alhawarat (2015) used a similar technique for extracting verses topics from the Qur'an.

As the candidates are considered as syntactic structures in the tree parser, For free word order and morphologically rich languages such as Arabic, morphology is essential in syntactic modelling [83]. [84] report that combining case feature into POS tagset is yield an optimal tagset for parsing Czech text. [85] report the case feature gives similar results for Arabic text. In our work, we investigate how considering morphological features affects the term candidate generation. A summary of most related work to Qur'anic concept identification is shown in Table 3.2. Further related work is described in Table B.3 in Appendix B.

This work is different from previous research in Arabic concept identification for the Qur'an as follows:

- This is the first concept identification method augmented with morphological analysis and internal information of the term in Arabic.
- Our method combines domain-specific knowledge and statistical knowledge for identifying important terms.

Research	**Unithood**	**Termhood**	**Coverage**
[77]	Only single-word terms	TF-IDF	Chapter no 12
[76]	Specific Patterns	TF-IDF	Whole Qur'an
[7]	Applied KP-Miner	Applied KP-Miner	Some parts and scopes

Table 3.2 A summary of the related work on Arabic Qur'anic

3.2.2 Hierarchies Extraction

Hierarchies of concepts are main semantic relations in any ontology [86], and they are the backbone of ontologies [6], [87] and the main components of ontologies [88], [89]. Hierarchical relations are important for many applications such as navigation and categorization. Hierarchical relations construction is a task of extracting pairs of terms linked by hypernym-hyponym ("is-a") relations.

Traditionally, hierarchies are created through a process of employing a group of experts [90], which is very expensive in terms of time and effort required. One of major challenge in ontology development is automatic discovery of taxonomies. Automatic discovery of "is-a" relations has been studied under different names such as concept hierarchies and taxonomy acquisition. Therefore, a number of approaches to automatically or semi-automatically extract hierarchies from textual data were developed.

3.2.2.1 Internal Structure Approaches

This approach relies on the constituents of compound terms, and is simple and highly effective [91]. However, this method cannot differentiate direct and indirect hypernyms [92]. Head-modifier based method relies on multi-word terms structure, which constitutes the majority of an ontology. It is very important for interpreting semantic relations and plays an important role in term formation [93]. This method groups multi-word terms into hyponymic relationships according to their head. The head of terms is called hypernym and the modifiers that belong to the same head are called hyponyms. For instance, from the Qur'an, the multi-word terms ("the day of Resurrection", " the day ofJudgment", "the day of shadow" and "the day of Approaching") have the same head, therefore they will be hyponyms of the term "Day".

Mukhtar et al. (2012) produced a dataset that contains 693 vocabulary items extracted from the second chapter of the Qur'an, known as the Vocabulary of Qur'anic Concepts (VOQ). The vocabulary items have been categorised into different categories based on head-modifier. The extracted vocabulary items are provided in sql and xml format. The authors used six different English translations of the Qur'an and applied a domain-independent tool called Termine from Frantzi & Ananiadou (1999) to extract the concepts.

Another related work for the Qur'an is in [66], which relies on a format of one English translation of the Qur'an that included some aspects such as Uppercase Letter. An uppercase letter is used to identify the concepts such as the Book. Another feature called compound noun is used to identify the relationship of hyponym and "Part-OF" between the concepts. A copula is used to identify the syntactic relationship between subject and adjective or noun. The ontology is based on the information obtained from domain experts. The development process is adopted from [66].

3.2.2.2 Lexical Patterns Approaches

This is also known as pattern-based, and is used to find hyponym relations from a large corpus. One example of this method is a Lexico-Syntactic pattern based on Hearst-patterns [94]. These patterns are made of two components: lexicalised patterns (e.g. "including", "especially" and "like"), or even POS tags (e.g. noun, verb and adjective). The second component is the chunk of the units that linked together in syntactical way such as noun phrase chunk tag (NP). The following patterns are the same patterns that were used for automatic hyponyms extraction in [94].

1. $NP_0\ such\ as\ NP_1, \ldots., NP_{n-1}\ (and|or) NP_n$
2. $Such\ NP_0\ as\ NP_1, \ldots., NP_{n-1}\ (and|or) NP_n$
3. $NP_1, NP_2, \ldots, NP_n\ (and|or) other\ NP_0$
4. $NP_0, (including|especially) NP_1, \ldots., NP_{n-1} (and|or) NP_n$

Lexico-Semantic pattern was introduced for the first time in [95], which is superior pattern-based approach to Lexico-Syntactic pattern in terms of efficiency and effectivity [96]. It allows using semantic information such as ontology elements in building pattern for extraction.

3.2.2.3 Clustering-based Approaches

In this approach, a vector of context is prepared for each term based on selected features. Then clustering is applied according to the calculated similarity measurement between terms vectors. This approach is useful to identify relations that do not explicitly occur in the text, but it is difficult to extract taxonomy, and fails to produce good results for a small corpus [92].

3.2.2.4 Graph-based Approaches

An example of this approach is [97] where each extracted term is added as a node. Then analysing the sentences that contain this node from the current corpus and the web produces weight relations between nodes. After that, they apply an algorithm to select best relation for nodes to be linked by.

3.2.2.5 Previous Work for The Qur’an

[7] is another work related to the Arabic text of part of the Qur’an which extract another types of relation such as “part of”, “kind of” and “synonym” based on association rules. [69] proposed an ontology extraction method for Arabic Wikipedia documents based on Wikipedia templates. [98] introduced a framework for Arabic ontology learning from textual resources. They defined a number of lexico-syntactical patterns using JAPE to model extracting noun phrases, instances, concepts and different type of relations such as “is-a” and “has-a” as shown in the Listing 3.1. The results were divided into two groups; in a first experiment by using Stanford syntactic parser on a limited number of sentence (34 sentence). Then, a second experiment covers 12779 words by using Arabic morphology analyser. The author could not parse the whole corpus due to it requiring all sentences to be well formed in terms of punctuation.

Listing 3.1

```
(syntaxnode.type=NP): instance
({Token.category=is-a}) // هو هي
(syntaxnode.type=NP) : concept
```

Another work related to developing Arabic ontology is [99], which focused on the domain of Arabic blogs in the computer technology domain. However this study did not mentioned whether the development was done manually or automatically.

Another work related to Arabic language but not to Qur’an is [9]. The author extracted a number of lexical patterns for Qur’an and newspaper and blog. [61] extracted the hierarchies for the terms of the first chapter of the English translation of the Qur’an manually based on Head-modifier. The extracted list was for the chapter no 2 of the Qur’an.

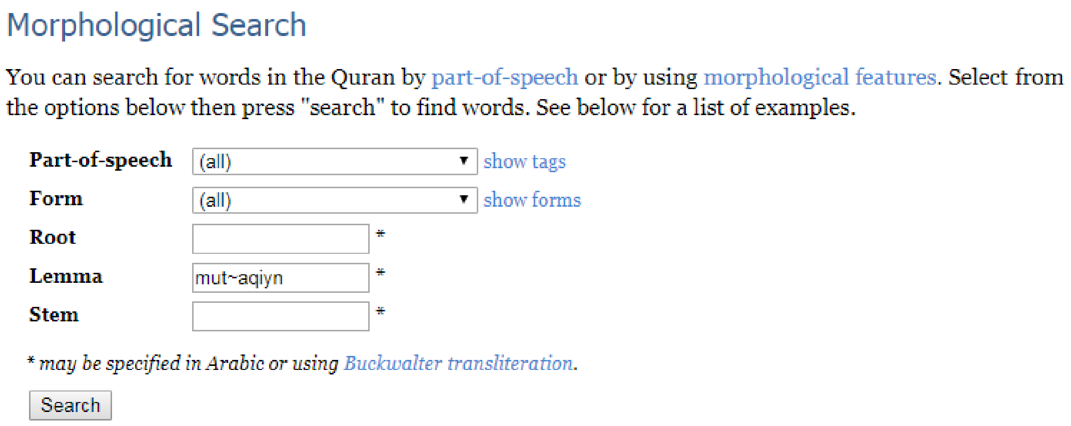
The nominal sentence in Arabic has a complex structure. The focus is on the copular sentence, which belongs to the nominal type of Arabic sentence [100]. A copula is a linguistic phenomenon that links a subject with a predicate in a sentence. Two types of copular constructs in Arabic; verbal and verbless. Verbal copular which known also as incomplete verbs are happened in (kaan). In verbless copula, some sentences do not require verb or copula to link the subject and the predicate [101]. In this case, there is a link called zero copula or null copula ∅. For example, "Khaled is a doctor" is written in Arabic as "خالد دكتور". Zero copula construct are definite nouns and their case is nominative[102]. Verbless copula also can be expressed by a pronoun. Examples from the Qur'an for these types of copula can be seen Table 3. Many examples and discussion for Arabic copula and its characteristics in [103], [104].

3.3 Annotation-Level Search For the Qur'an

An annotation-level search is a way of processing the content of corpus. It allows user to make queries at the level of the annotations. In other words, allowing users to search for a specific or a number of tags instead of the collection text. A number of software that provides Corpus Query Language (CQLs) can be found in [105]. However, most of these software were made for a specific language or a limited to the specific collection of text.

A few attempts have been made for available Arabic resources. A dictionary and morphological search tool was made free available online by [21]. This feature allowed annotators to search over tag like POS and root in order to find the correct occurrences of the targeted meaning. However, they only allow searching over a certain group of morphological features such as POS, form , ROOT, Lemma and Stem based on Buckwalter[1] [106] transliteration and does not allow searching using Arabic letters.

[1] Is an orthographic transliteration scheme uses ASCII characters for representing Arabic text for computers and to be easily read by non-Arabic researchers.



Morphological Search

You can search for words in the Quran by part-of-speech or by using morphological features. Select from the options below then press "search" to find words. See below for a list of examples.

Part-of-speech: (all) show tags
Form: (all) show forms
Root: *
Lemma: mut~aqiyn *
Stem: *

* *may be specified in Arabic or using Buckwalter transliteration.*

Figure 3.11 Morphological search by [21]

Another attempt reported by [107] in her MSc dissertation unified a number of Qur'anic datasets and offered an annotation-based search using sketch engine. The author made a powerful tool that allowed the user to extracted knowledge based on the annotation in three different datasets (Qurany, QurAna and QAC).

3.4 Summary

This chapter has reviewed relevant work to this thesis in three different parts: existing Qur'anic ontologies and annotations, ontology learning from textual resources and annotation-level search. The first part compared existing Qur'anic ontologies according to a range of appropriate criteria including an analysis of the limitation of previous ontologies work for the Qur'an. The results shown that only few ontological annotations have been made for the entire the Qur'an. In addition, there are a variation in the format used for encoding these annotations.

Following the review of existing ontologies, concept identification approaches with great focus on Arabic language and the Qur'anic text was included. The literature of concept identification has revealed an important knowledge about the current state of Arabic concept identification. There is a big gap between the improvement in concept identification and the current works in concept identification from Arabic text. Most work for concept identification from text of Arabic applied weighting scheme regardless the size of the text or its specifications for measuring term relevance. In addition, hierarchical relations extraction methods for the Arabic and Qur'anic domain was also reviewed.

Part II

Collecting and Discovering Qur’anic Resources

Chapter 4

Combining and Extracting Qur'anic Knowledge: The AQD Arabic Qur'anic Database.

One of the main reasons for the lack of Arabic representation in the field of semantic web is the lack of Arabic information extraction tools, also one of the main problems for Arabic ontology [8]. Part one of the previous chapter, Chapter 3, addressed the issue that most Qur'anic annotations have not been done for the entire the Qur'an, and there is no clear consensus on the used format. This chapter; presents a new work for available Qur'anic textual annotations called the Arabic Qur'anic Database (AQD). It merges several annotations from different formats and resources, including morphological, structural, chronological and ontological. In addition, it provides an annotation-based search that draws on all available resources. This combination of different resources and search tool has not been done before for the domain of the Qur'an. Creating such an environment will enable researchers in Arabic computational linguistics and Qur'anic research to investigate new directions and new features for NLP learning tasks. The aim is to make use of these available annotations and integrate them in one single query. This will help researchers in Islamic studies discover hidden knowledge and mine Qur'anic information. It is also a helpful resource for researchers of classical Arabic grammar in that they can search for features from different resources for certain grammatical relations or patterns. We show statistical analysis of the combined annotations and examples of grammatical relations and linguistic phenomena that occur in the Arabic as well as how to make queries to extract them.

The contributions presented in this chapter are two-fold. First, different Qur’anic resources collected from different annotations and formats are combined. Second, an annotation-based search for making a query that integrates these different annotations in a single query is designed and implemented. In addition, this chapter offers a new environment for discovering and mining hidden Qur’anic knowledge. This chapter is based on our published paper [108].

4.1 Combining Qur’anic Annotations

Four different types of annotations have been combined into the MySQL Database, which is a Relational Database Management System (RDBMS). The selection of the annotations was based on the criteria studied in section 3.1.1, such as the language of the text and coverage proportion and availability. We chose only annotations that include Arabic text, cover all the entire Qur’an and are available for free. Therefore, the morphological layer is based on annotations from [109], which includes segments, the basic units that comprise the text of the Qur’an. For the topics, Qurany, a set of annotation by [48], is also available for the topic of verses. Another dataset collected from QAC includes verses and lists of concepts. The semantic layer [41] represents semantic tags for personal pronouns. In addition, chronological information and orders are also included based on [110], [111].

4.1.1 Morphological

The morphological annotations contain lexical information, which includes the diacriticised and undiacriticised formats of Arabic, Buckwalter transliteration, POS tag, root, lemma, and clitic information.

Inflectional information includes person, gender, number, voice, aspect, mood and case. The derivational information includes the form and whether each segment is active or passive.

4.1.2 Structural

The structural annotations are the relations between the divisions and text parts of the Qur'an. For example, the relationship between chapters and verses, verses and words, or words and segments. In addition, this includes the relations between Juz and Hizb divisions. All these relations and entities are included in the database. This kind of annotations had been used for research in chronological analysis as well as it is useful for ontology based on the Qur'anic structure.

4.1.3 Chronological Annotations

Five types of orderings are included in the AQD database. Figure 4.3 outlines five types of orders for each verse in the Qur'an. *order5* is the order of verses according to their locations in the 114 chapters. This order starts from 1 and ranges to 114. The second order, *order6*, is the order of verses according to their revelation place, either Makkah or Madinah. It contains only two notations, 1 for verses revealed in Makkah and 2 for verses revealed in Madinah. *order2* contains the 194 blocks of verses according to their revelation times; for example, block 1 contains the verses from 1 to 5 of chapter 96. *order3* is an order proposed by Mehdi Bazargan (d. HS 1373/1995) and divides the verses into 22 groups. *order4* is called the modified Bazargan order, and was obtained by combining some consecutive phases of the Bazargan to produce seven phases [110].

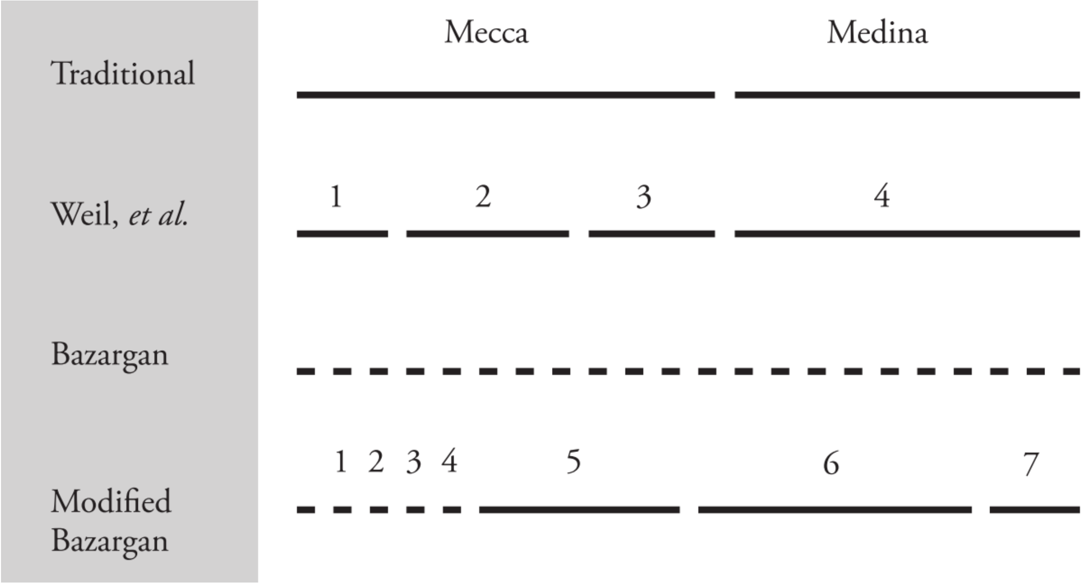


Figure 4.1 Different chronological orders of the Qur’anic verses obtained from [110]

4.1.4 Ontological

Four ontological annotations have been included in this database. This database includes the dataset of QAC ontology, which is composed of 294 concepts linked to each other hierarchically with Qur’anic words. It also cover the Qurany datasets, which is made of topics that linked to each other and the verses of the Qur’an. QurAna , which has several referents linked to personal pronoun segments, is also included in this database.

4.1.5 Mapping Different Segmentation Based Found in QAC and QurAna

Both morphological annotations from QAC and pronominal information from QurAna deal with the segment layer of the Qur’an. QAC gives features like POS tag, lemma and root for each segment, while QurAna gives the referent information for only the personal pronoun segment. The problem with using two datasets is that they do not have the same number of segments and they use different spelling system of the

Qur’an. Therefore, the segments are not identical in both datasets. There are 128,219 segments in QAC and 127,795 in QurAna. This appears to be only a 424 segment difference, but this becomes a more significant problem when we consider the variations in segmentation systems in each dataset. Some words are segmented into two parts in QAC but considered one part in QurAna (i.e. (“يظلمون” “doing wrong”), (“تظلمون”, “be wronged”) and (“ياأيها”, “O you”)). On the other hand, some segments are tagged with a referent in QurAna when they are not pronouns, such as (“ذلك”, “that for masculine singular nouns”) and (“تلك”, “that for famine singular or plural nouns”). This makes the problem worse because we must either remove segments from QAC or add segments to QurAna.

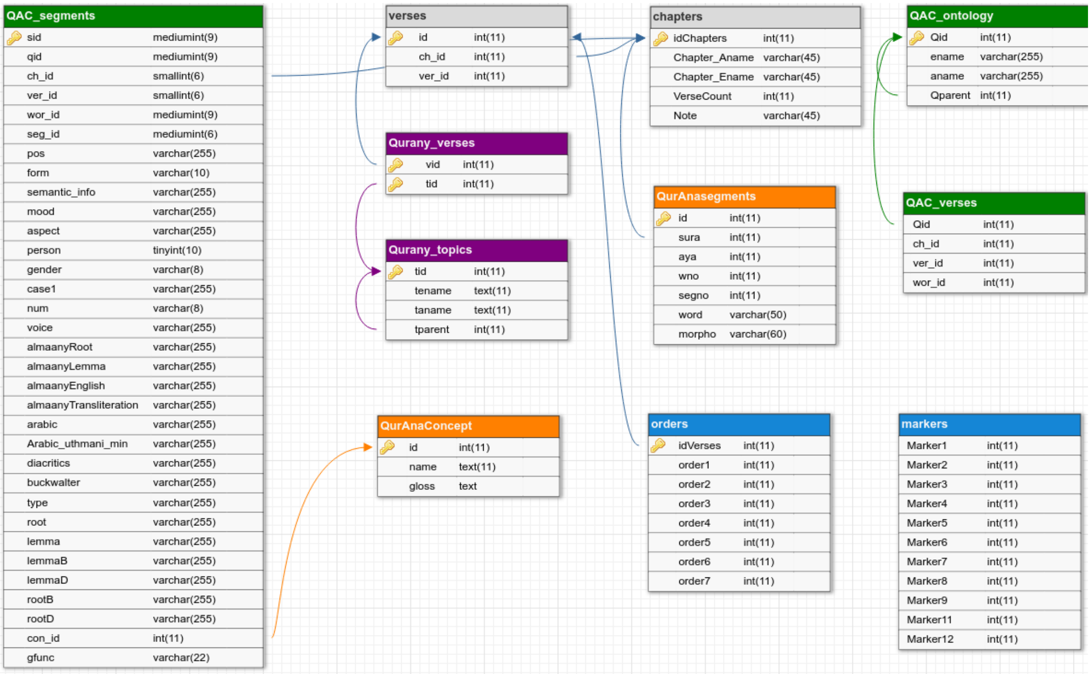
We used the index of location notation, as the two datasets are indexed by location notation (chapter, verse, word, segment) for every segment. This allowed us to retrieve all relevant information for a single segment using the SQL query below in Figure 4.2. We only focused on segments tagged with PRON (indicating a pronoun) because the only addition to our database from QurAna is the foreign key for the referent number of a pronoun.

```
SELECT qac_segments.sid, qurana_segments.id, qac_segments.diacritics,
     qurana_segments.word
FROM   qac_segments, qurana_segments
WHERE  qac_segments.ch_id =qurana_segments.sura
AND    qac_segments.ver_id = qurana_segments.aya
AND    qac_segments.wor_id = qurana_segments.wno
AND    Trim(qac_segments.pos) = "PRON"
AND    Trim(qurana_segments.morpho) LIKE "%PRON%"
AND    Trim(qac_segments.arabic) = Trim(qurana_segments.word)
```

Figure 4.2 SQL query was used for retrieving two aligned datasets QurAna and QAC

4.1.6 The Database Schema

Figure 4.3 depicts the scheme of the Arabic Qur'anic Database (AQD), which combines knowledge about the Qur'an from several different resources. The figure depicts the multiple tables in the database. Tables are coloured according to source. Some of tables are self-referential, like Qurany_topics, which has topics (some of which are linked to verses) and subtopics. QAC-ontology is also composed of concepts (some of which are linked to words) and their sub-concepts. Chronological annotations are imported from our previous research [111], which investigated seven types of markers obtained from the Qur'anic verses against five type of divisions.



QAC_segments	
sid	mediumint(9)
qid	mediumint(9)
ch_id	smallint(6)
ver_id	smallint(6)
wor_id	mediumint(9)
seg_id	mediumint(6)
pos	varchar(255)
form	varchar(10)
semantic_info	varchar(255)
mood	varchar(255)
aspect	varchar(255)
person	tinyint(10)
gender	varchar(8)
case1	varchar(255)
num	varchar(8)
voice	varchar(255)
almaanyRoot	varchar(255)
almaanyLemma	varchar(255)
almaanyEnglish	varchar(255)
almaanyTransliteration	varchar(255)
arabic	varchar(255)
Arabic_uthmani_min	varchar(255)
diacritics	varchar(255)
buckwalter	varchar(255)
type	varchar(255)
root	varchar(255)
lemma	varchar(255)
lemmaB	varchar(255)
lemmaD	varchar(255)
rootB	varchar(255)
rootD	varchar(255)
con_id	int(11)
gfunc	varchar(22)

verses	
id	int(11)
ch_id	int(11)
ver_id	int(11)

chapters	
idChapters	int(11)
Chapter_Aname	varchar(45)
Chapter_Ename	varchar(45)
VerseCount	int(11)
Note	varchar(45)

QAC_ontology	
Qid	int(11)
ename	varchar(255)
aname	varchar(255)
Qparent	int(11)

Qurany_verses	
vid	int(11)
tid	int(11)

Qurany_topics	
tid	int(11)
tename	text(11)
taname	text(11)
tparent	int(11)

QurAnasegments	
id	int(11)
sura	int(11)
aya	int(11)
wno	int(11)
segno	int(11)
word	varchar(50)
morpho	varchar(60)

QAC_verses	
Qid	int(11)
ch_id	int(11)
ver_id	int(11)
wor_id	int(11)

QurAnaConcept	
id	int(11)
name	text(11)
gloss	text

orders	
idVerses	int(11)
order1	int(11)
order2	int(11)
order3	int(11)
order4	int(11)
order5	int(11)
order6	int(11)
order7	int(11)

markers	
Marker1	int(11)
Marker2	int(11)
Marker3	int(11)
Marker4	int(11)
Marker5	int(11)
Marker6	int(11)
Marker7	int(11)
Marker8	int(11)
Marker9	int(11)
Marker11	int(11)
Marker12	int(11)

Figure 4.3 The schema of the database

Figure 4.3 shows the AQD database tables implemented as a set of MySQL related tables that store different available types of information for the Qur'an, such as morphological, topical, conceptual, structural and chronological.

4.2 Aligning Qur'anic Ontologies

An experiment of aligning two ontologies for the Qur'an was done on Qurany and QAC ontologies. The aim of this experiment is to find similar entities distributed in Qur'anic ontologies and to investigate the possibility of hierarchical relation over alignment.

4.2.1.1 Ontology Alignment

To extract corresponding concepts from two ontologies let us assume we have two ontologies. Let x be the first ontology and y be the second ontology that we want to learn the relations from. A is a list of alignments which is the output of algorithm of matching the entities of two ontologies $x\ and\ y$. The alignment list consists of tuples such as (x_i, y_i, w) where x_i is a term from x ontology and y_i is a term from y ontology and w is the weight calculated using similarity measure method such as formula in (4.1). Terms that represent same meaning in the alignment list are called corresponding terms.

4.2.1.2 Similarity Measurements

In order to decide two terms are similar, there are several approaches for measuring. We will apply the following hybrid matching comparison. This approach takes advantage of combining Fuzzy Bilingual Lexical-based and structure-based methods for aligning highly variants ontologies. It is aggregating multiple similarity scores for a given pair of concepts into a single value as it shown in Equation (4.1)**Error! Reference source not found.**.

$$HB(a, b) = max(Lex_{AR}, Lex_{EN}, SB) \tag{4.1}$$

Whereas Lex_{AR} is the score of lexical-based match using Arabic language concept labels. Lex_{EN} is the score of lexical-based match using English translation of concept labels. SB is the score obtained by structure-based match using instances belonging to the given pairs regardless their labels.

4.2.1.3 Fuzzy Lexical-based Matching of Bilingual labels of Concepts

We will refer to the fuzzy lexical base of the Arabic labels as Lex_{AR} and for English labels as Lex_{EN}.

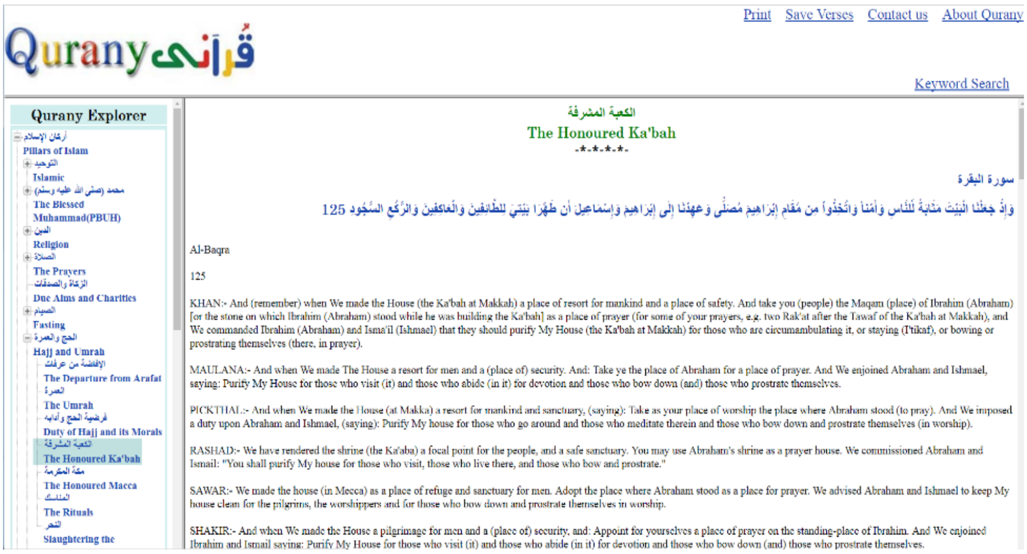
Fuzzy bilingual lexical-based was modelled using Dice's coefficient adapted from [112] as shown in Equation (4.2).

$$Lex(a,b) = \frac{|a \ \cap b|}{\left(\frac{|a| + |b|}{2}\right)} \qquad (4.2)$$

Where $a\,nd\,b$ are sets of bigrams for the matched labels. This method detects the common set between the given pairs of bigram sets. For example the pairs "Umra" and "The Umrah" have a set of {"Um", "mr", "ra"} in common in their set of bigrams, which gives a similarity of 60% between them.

4.2.1.4 Structure-based Matching

This method takes into account the occurrences of a concept as a feature for indicating the similarity. It takes all instances that are found as children for the given pair that are going to be matched. Both resources have been linked with a number of verses they were mentioned in. Figure 4.4 and Figure 4.5 show an example of the concept "Ka'bah" and its occurrences in these resources.



الكعبة المشرفة
The Honoured Ka'bah
-*-*-*-*-

سورة البقرة

وَإِذْ جَعَلْنَا الْبَيْتَ مَثَابَةً لِّلنَّاسِ وَأَمْنًا وَاتَّخِذُواْ مِن مَّقَامِ إِبْرَاهِيمَ مُصَلًّى وَعَهِدْنَا إِلَى إِبْرَاهِيمَ وَإِسْمَاعِيلَ أَن طَهِّرَا بَيْتِيَ لِلطَّائِفِينَ وَالْعَاكِفِينَ وَالرُّكَّعِ السُّجُودِ 125

Al-Baqra

125

KHAN:- And (remember) when We made the House (the Ka'bah at Makkah) a place of resort for mankind and a place of safety. And take you (people) the Maqam (place) of Ibrahim (Abraham) [or the stone on which Ibrahim (Abraham) stood while he was building the Ka'bah] as a place of prayer (for some of your prayers, e.g. two Rak'at after the Tawaf of the Ka'bah at Makkah), and We commanded Ibrahim (Abraham) and Isma'il (Ishmael) that they should purify My House (the Ka'bah at Makkah) for those who are circumambulating it, or staying (I'tikaf), or bowing or prostrating themselves (there, in prayer).

MAULANA:- And when We made The House a resort for men and a (place of) security. And: Take ye the place of Abraham for a place of prayer. And We enjoined Abraham and Ishmael, saying: Purify My House for those who visit (it) and those who abide (in it) for devotion and those who bow down (and) those who prostrate themselves.

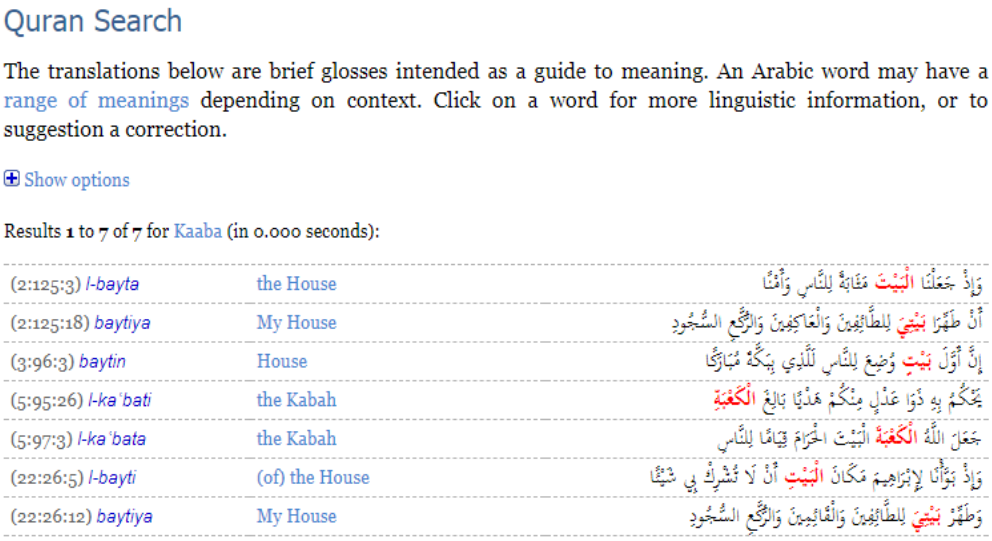
PICKTHAL:- And when We made the House (at Makka) a resort for mankind and sanctuary, (saying): Take as your place of worship the place where Abraham stood (to pray). And We imposed a duty upon Abraham and Ishmael, (saying): Purify My house for those who go around and those who meditate therein and those who bow down and prostrate themselves (in worship).

RASHAD:- We have rendered the shrine (the Ka'aba) a focal point for the people, and a safe sanctuary. You may use Abraham's shrine as a prayer house. We commissioned Abraham and Ismail: "You shall purify My house for those who visit, those who live there, and those who bow and prostrate."

SAWAR:- We made the house (in Mecca) as a place of refuge and sanctuary for men. Adopt the place where Abraham stood as a place for prayer. We advised Abraham and Ishmael to keep My house clean for the pilgrims, the worshippers and for those who bow down and prostrate themselves in worship.

SHAKIR:- And when We made the House a pilgrimage for men and a (place of) security, and: Appoint for yourselves a place of prayer on the standing-place of Ibrahim. And We enjoined Ibrahim and Ismail saying: Purify My House for those who visit (it) and those who abide (in it) for devotion and those who bow down (and) those who prostrate themselves.

Figure 4.4 Qurany project navigating for the concept "Ka'bah"



Quran Search

The translations below are brief glosses intended as a guide to meaning. An Arabic word may have a range of meanings depending on context. Click on a word for more linguistic information, or to suggestion a correction.

Results 1 to 7 of 7 for Kaaba (in 0.000 seconds):

(2:125:3) *l-bayta*	the House	وَإِذْ جَعَلْنَا الْبَيْتَ مَثَابَةً لِلنَّاسِ وَأَمْنًا
(2:125:18) *baytiya*	My House	أَنْ طَهِّرَا بَيْتِيَ لِلطَّائِفِينَ وَالْعَاكِفِينَ وَالرُّكَّعِ السُّجُودِ
(3:96:3) *baytin*	House	إِنَّ أَوَّلَ بَيْتٍ وُضِعَ لِلنَّاسِ لَلَّذِي بِبَكَّةَ مُبَارَكًا
(5:95:26) *l-ka'bati*	the Kabah	يَحْكُمُ بِهِ ذَوَا عَدْلٍ مِنْكُمْ هَدْيًا بَالِغَ الْكَعْبَةِ
(5:97:3) *l-ka'bata*	the Kabah	جَعَلَ اللَّهُ الْكَعْبَةَ الْبَيْتَ الْحَرَامَ قِيَامًا لِلنَّاسِ
(22:26:5) *l-bayti*	(of) the House	وَإِذْ بَوَّأْنَا لِإِبْرَاهِيمَ مَكَانَ الْبَيْتِ أَنْ لَا تُشْرِكْ بِي شَيْئًا
(22:26:12) *baytiya*	My House	وَطَهِّرْ بَيْتِيَ لِلطَّائِفِينَ وَالْقَائِمِينَ وَالرُّكَّعِ السُّجُودِ

Figure 4.5 Qur'anic Arabic Corpus Search for the concept "Ka'bah"

Figure 4.4 shows the concept of "Ka'bah" from [48], the Qurany project[1], which has been expressed in different labels in both Arabic and English translation in comparison with the same concept in Figure 4.5. Figure 4.5 shows the occurrences of the same concept used above. Note,

[1] http://quranytopics.appspot.com/

we extracted the occurrences of the concept of QAC from the list of Qur'anic topics[2] in which all concepts are attached with where they were mentioned in the Qur'an based on chapter, verse and word numbers. Figure 4.6 depicts a single concept with their occurrences set based on two different ontologies. Concept X in Qurany occurs in a set of verses $\{132, 389, 390, 764, 766, 2621\}$, while the concept Y in QAC occurrences vector is$\{132, 132, 390, 764, 766, 2621, 2621\}$. Although both Arabic and English translations labels of this pair are not matched, the size of intersection between their instances is high which is an indication that they represent the same concept. This pair of concepts are clearly sharing many verses, which can be modelled by the intersection of X and Y. Thus, the larger intersection size means the two concepts are similar.

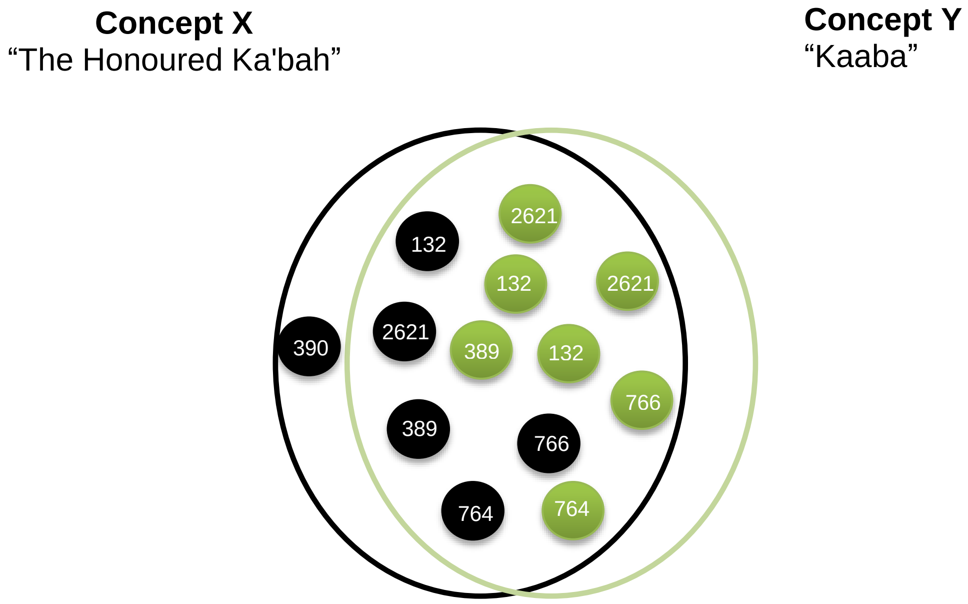


Figure 4.6 The Venn diagram of a concept and its occurrences in different ontologies

In order to compute the similarity, every concept has been extracted and given a unique id. Then for each concept, we add all instances (occurrences of these concepts in the Qur'an) in a vector. After that, we

[2] http://corpus.quran.com/topics.jsp

applied Jaccard similarity measures [113] using Equation (4.3), which is one of the most common methods for computing similarity based on sample sets. To compute similarity between two sets, it is the ratio of their intersection divided by their union.

$$SB(a, b) = \frac{|a \cap b|}{|a \cup b|} \qquad (4.3)$$

For example the Jaccard similarity for the above example is:

The intersection is $|a \cap b| = \{132, 2621, 389, 764, 766\}$ and the union is$|a \cup b| = \{132, 389, 390, 764, 766, 2621\}$. So the SB for this example is:$SB(a, b) = \frac{|a \cap b|}{|a \cup b|} = \frac{5}{6} = 0.833$.

4.2.1.5 Aligning Qurany and QAC ontologies

An experiment on aligning available Qur'anic ontologies have been conducted on Qurany and QAC using a number of similarity measures. The labels of concepts in Arabic and English were obtained from both ontologies as well as the information of verses that linked under these concepts.

Lex_EN	Lex_AR	SB	HB	QAC EN	Qyrany EN
85.71	100	30.952	100	Pharaoh	Pharaoh
55.55	71.42	100	100	Sabians	The Sobians
28.57	100	60	100	Jibreel	Gabriel
80	100	100	100	Marut	Marut
80	100	100	100	Harut	Harut
39.39	100	87.5	100	Masjid al-Haram	The Most Sacred Mosque in Makka)
61.53	75	100	100	Umra	The Umrah
61.53	100	2	100	Hell	Hell Fire
0	100	0	100	Musa	Moses
55.55	71.42	100.00	75.66	Musa	Moses

Table 4.1 A sample of results of the four compared measures

Table 4.1 shows a sample of obtained results by the four selected measure methods. Lex_EN is the fuzzy lexical-based for English translation, Lex_AR is the fuzzy of lexical-based for Arabic, SB is the structure-based, and HB is the new method combines all of them. Every algorithm gives a sorted list of pairs from the most likely similar to least likely. Table 4.3 shows more results. Results made publicly available for as training data or in evaluation other approaches[3]. These results in both tables are ranked based on HB method.

As all selected methods of similarity produce alignment as a ranked list, we used Average precision (AvP), to evaluate ranked returned list. AvP is commonly applied in ranked-based extraction such as in [69], [26]. AvP requires the retrieved pairs to be validated, therefore the researchers have manually validated the top 100 returned pairs of the fourth methods with 1 for pairs that correctly returned and 0 for the rest.

The equation of AvP is shown in Equation (**Error! Reference source not found.**) and the top-50 ranked pairs is in Table 4.2.

$$AvP = \frac{\sum_{k=1}^{n}(p(k) \times rel(k))}{n_c} \qquad (4.4)$$

Where $p(k)$ is the precision at cut-off k in the pairs list, n means the size of the ranked list, n_c is the total number of relevant pairs that were returned by the method and $rel(k)$ is a binary function that indicated whether or not the retrieved pairs are similar. The output of $rel(k)$ is 1 if a $concept_k$, which means the concept at k, the pairs are same. Otherwise $rel(k)$ is 0.

Similarity Measures	Recall	Precision	AvP
Lex_{EN}	0.776	0.760	0.883
Lex_{AR}	**0.833**	0.900	0.983

[3] http://salrehaili.com/QuranOntology/Alignments

SB	0.714	0.60	0.647
HB	0.721	**0.980**	**0.980**

Table 4.2 Methods comparision based on recall, precision and AvP for top-50

Table 4.2 shows the results of our experiment on aligning Qur'anic ontological annotations based on four alignment methods for the first 50 pairs. The results have shown that the better results was achieved with HB in terms of precision and AvP, while the lexical-based match for Arabic labels obtained the highest recall.

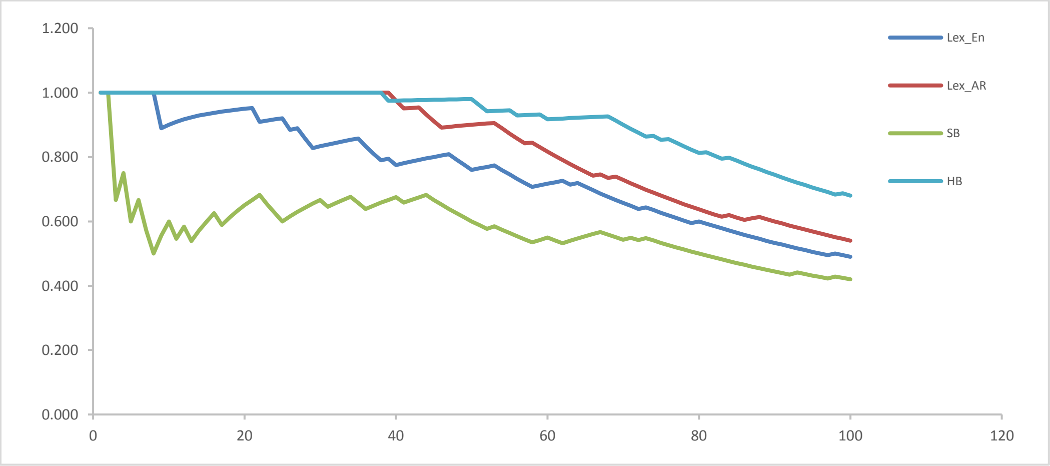


Figure 4.7 The precision over top-100 returned pairs

Figure 4.7 depicts the relationship between the precision over the top 100 returned pairs. The figure clearly shows that HP has outperforms other algorithms. Significantly, lexical-based for Arabic has obtained similar results but it decreases sharply after number 40 in the ranked list. Over all the new algorithm HB has outperforms the three compared algorithm.

Although fuzzy-based performs better than exact lexial-based, its effect only on labels expressed like ("Firdous", "The Firdous"), ("Qaroun", "Qarun"). Some concepts were incorrectly aligned by this method such as ("Harut", "Marut") and some concepts in Arabic like ("الجن" and "جنة"). This is because they have the same spelling in Arabic and different meaning. Structure-based was able to return a number of pairs correctly regardless their labels such as ("Thamud", "Salih People") and ("Kaaba", "The Honoured Ka'bah"). However, not all concept in QAC are provided with verses that they occur in. And this is the reason why structure-based evaluation had such a poor result.

Our approach performs better in ontologies where their entities have been labelled using more than one language. In addition, it works to enrich ontology relations compared to the work in Chapter 6.

Lex_EN	Lex_AR	SB	HB	QAC EN	QAC AR	Qyrany EN	Qyrany AR
80.00	100.00	2.56	100	Allah	الله	Allah	الله
69.23	100.00	0.00	100	Christianity	النصارى	The Christians	النصارى
90.00	100.00	0.00	100	Paradise	الجنة	The Paradise	الجنة
66.66	100.00	0.00	100	Angel	الملائكة	The Angels	الملائكة
38.09	100.00	0.00	100	Satan	الشيطان	the Devil Satan	الشيطان
85.71	100.00	30.95	100	Pharaoh	فرعون	Pharaoh	فرعون
55.55	71.42	100.00	100	Sabians	الصابئين	The Sobians	الصابئون
28.57	100.00	60.00	100	Jibreel	جبريل	Gabriel	جبريل
80.00	100.00	100.00	100	Marut	ماروت	Marut	ماروت
80.00	100.00	100.00	100	Harut	هاروت	Harut	هاروت
39.39	100.00	87.50	100	Masjid al-Haram	المسجد الحرام	Al-Masjid Al-Haram(The Most Sacred Mosque in Makka)	المسجد الحرام
61.53	75.00	100.00	100	Umra	عمرة	The Umrah	العمرة
61.53	100.00	2.00	100	Hell	جهنم	Hell Fire	جهنم
0.00	100.00	0.00	100	Musa	موسى	Moses	موسى
0.00	100.00	0.00	100	Musa	موسى	Moses	موسى
20.00	100.00	5.00	100	Harun	هارون	Aaron	هارون

12.50	100.00	100.00	100	Injeel	الإنجيل	The Gospel	الإنجيل
0.00	100.00	100.00	100	Torah	التوراة	The Bible	التوراة
42.85	100.00	0.00	100	Ibrahim	إبراهيم	Abraham	إبراهيم
42.85	100.00	0.00	100	Ibrahim	إبراهيم	Abraham	إبراهيم
94.73	100.00	2.00	100	Day of Resurrection	يوم القيامة	Day of Resurrection	يوم القيامة
0.00	100.00	42.86	100	Al-Jahiliyah	الجاهلية	The Paganism	الجاهلية
0.00	100.00	0.00	100	Nuh	نوح	Noah	نوح
0.00	100.00	0.00	100	Nuh	نوح	Noah	نوح
66.66	100.00	58.49	100	Jinn	الجن	The Jinn	الجن
15.38	100.00	100.00	100	Garden of Eden	جنات عدن	Adn Paradise	جنات عدن
0.00	100.00	5.56	100	Yaqub	يعقوب	Jacob	يعقوب
24.24	100.00	5.56	100	Companions of the Cave	أصحاب الكهف	Cave People	أصحاب الكهف
48.00	100.00	18.75	100	Dhul Qarnayn	ذو القرنين	Dhul-Quarnain	ذو القرنين
92.30	100.00	25.00	100	Gog and Magog	يأجوج ومأجوج	Gog and Magog	يأجوج ومأجوج
53.33	100.00	100.00	100	Magians	المجوس	The Magi	المجوس
85.71	100.00	50.00	100	Firdous	الفردوس	Firdous	الفردوس
22.22	100.00	50.00	100	Companions of the Rass	أصحاب الرس	Ar-Rass People	أصحاب الرس
47.05	100.00	0.00	100	Sheba	سبأ	Saba'(Sheba)	سبأ
54.54	100.00	80.00	100	Qarun	قارون	Qaroun	قارون
75.00	100.00	25.00	100	Romans	الروم	The Romans	الروم
0.00	75.00	100.00	100	Zaqqum	زقوم	Infernal Tree	الزقوم
36.36	100.00	100.00	100	Malik	مالك	Maleck	مالك
50.00	0.00	100.00	100	Tubba	تبع	People of Tubba	قوم تُبَّع

Table 4.3 More results of the four compared measures

Annotations	Number	Percentage
Morphological	1,410,409	82.17%
Structural	212,088	12.36%
Chronological	31,180	1.82%

Pronominal	25,528	1.49%
Topical	19,099	1.11%
Conceptual	18,075	1.05%

Table 4.4 Distribution of the Qur'anic annotations included in AQD

The AQD comprises 1,716,379 entries, about 1,410,409 of which are morphological. The morphological tags are assigned to every segment. Note that the number of segments is about 128,219 as mentioned above and the number of tagged values are 1,716,379. So, the number of morphological includes the number of morphology features for each segment not only the number of segments. There are 212,088 structural entries, 128,219 segments assigned to words, 77,429 words linked with verses, 6,236 verses linked with chapters, 114 chapters linked with Hizb, 60 Hizb linked with Juz and 30 Juz. There are 31,180 chronological entries divided into five types of chronological orders, and each verse is assigned to one of these different orders. There are 19,099 topical entries, each linked to subtopics and verses. Single-word concepts from QAC are also included, and they are 18,075 entries including concepts and their sub-concepts, as well as concepts and words.

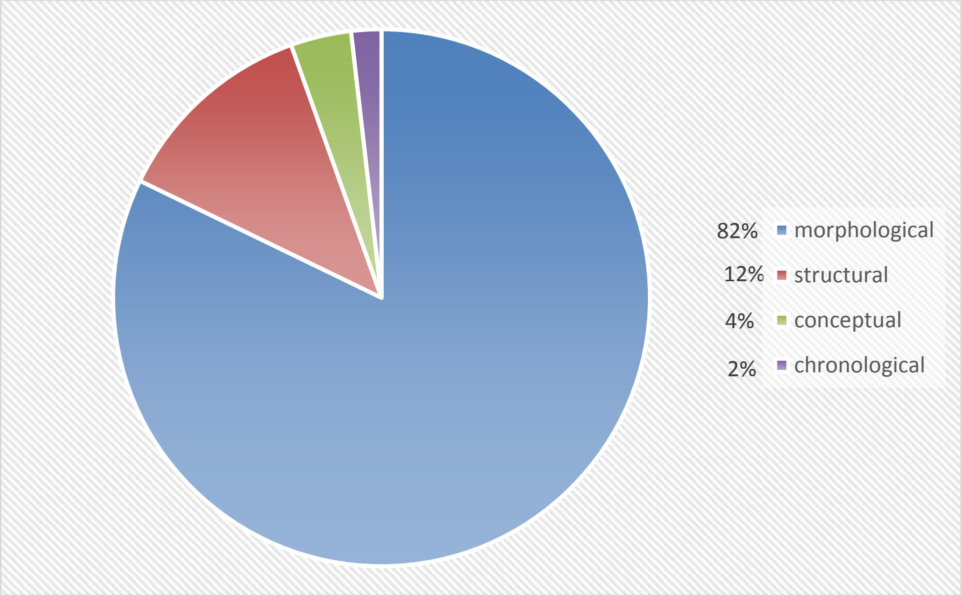


Figure 4.8 Comparing different types of Qur'anic annotations in AQD

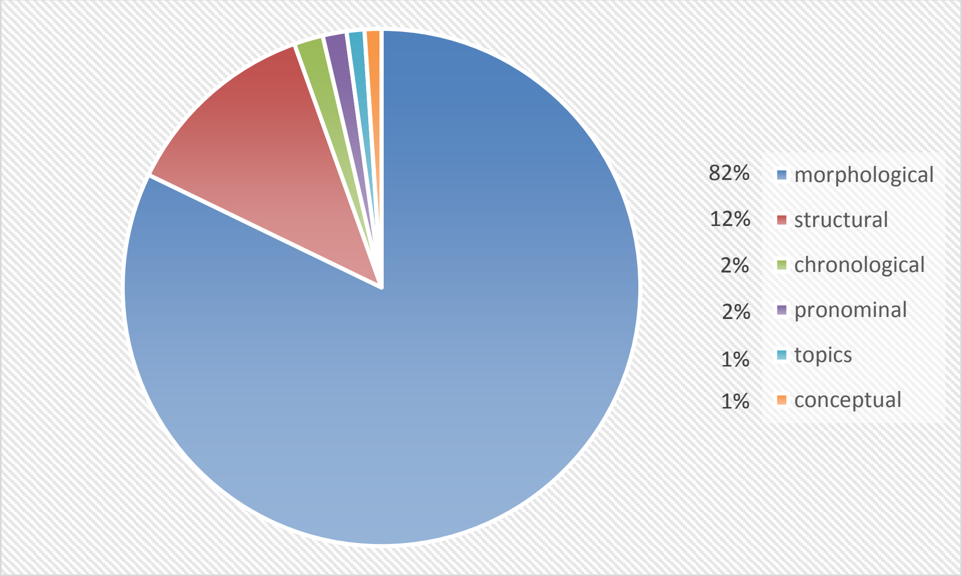


Figure 4.9 Distribution of Qur'anic annotations in AQD

Figure 4.8 and Figure 4.9 show the distributions of morphological annotations, which vary because they are at the segment layer. However, other semantic annotations with the same layer have fewer of annotations, such as pronoun references. This clearly shows the need for more semantic annotations for the Qur'anic text.

4.2.2 Labels of The Combined Annotations

Table 4.5 lists all included features from the different resources that can be used in the annotation-based search. The first column is the annotation type, including five different annotations. The second column provides the labels that should be used in the query as attributes. The third column gives examples of the values of the attributes. All attribute labels are case sensitive and can be used in the query as listed. INFLNV indicates inflectional qualities of nominals and verbs and INFLV indicates inflectional qualities of verbs. INFLN denotes inflectional qualities of nominals and DERIV indicates derivational features.

Annotation type	**Label**	**Values**
Lexical	diacritics	The Diacriticised format of the segment.
	buckwalter	Buckwalter transliteration of the segment.
	arabic	Undiacriticised Arabic uthmanic format.
	lemmaD	The Diacriticised for of the lemma.
	lemmaB	The Buckwalter form of lemma.
	rootD	The Diacriticised form of root
	rootB	The Buckwalter form of root.
	pos	Based on QAC tagset
INFLNV	person	{1, 2, 3}
	gender	{M, F}
	number	{S, D, P}
INFLN	case1	{nom, gen, acc}

	det	The determiner.
INFLV	aspect	{imperfect, imperative, perfect}
	voice	{active, passive}
	mood	{jussive, subjunctive}
	type	The clitic information
DERIV	form	{II, III, IV, V, VI, VII, VIII, IX, X, XI, XII}
	Active participle	
Ontological	Qurany	{1-1135} can be found on Qurany website
	QAC	{1-294} can be found on QAC website
	QurAna	{1-1028}
Chronological	order2	{1-194}, the blocks
	order3	{1-22}
	order4	{1-7}
	order5	{1-114}
	order6	{1-2}
Structural	chapter	{1-114}
	ver_id	Start from 1 for each chapter.
	vid	A representation for all verses {1-6236}
	Juz	{1-30}
	Hizb	{1-60}

Table 4.5 Labels of the multiple resources for annotations

4.3 Annotation Based Search

To make use of the combined annotations, we built a sequential search tool shown in Figure 4.10. This tool allows users to extract concordance lists of a given complex query based on different features. The query is received by the system, comprising two components: parsing the query into nodes and transferring the nodes to SQL query format. The execution manages the implementation of the SQL query in a sequence as an array of linked lists. Finally, the relevant instances, which meet all given conditions in the query, are stored in text files.

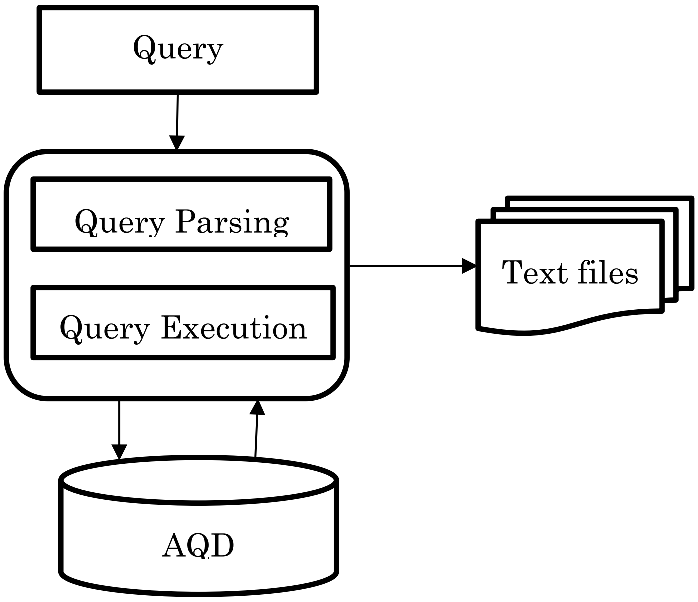


Figure 4.10 The Complex query implementation diagram

4.3.1 Query Syntax

Extended Backus-Naur forum (EBNF), one of most common context-free grammars for notation techniques, is used here to describe the query in detail. It is often used to describe how compilers and computer languages work. EBNF is used because we are presenting a language for a query that has different brackets and operators instead of words and letters. Natural language can be described using another language that has regular expression and that deals with letters. However, this is not a natural language; it is made of different brackets and operators.

```
Patterns            =   “{”, {sequences}, “}”
Sequences           =   “[”, [Node] , [logical_operator], “]”,
                        [Wild_card]
Node                =   {Attribute, Match_operator, Value}
Logical_operator    =   “&” | ”|”
Wild_card           =   “*” | “+” “?”
Match_operator      =   “==” | “!=” | “re” | “not re” | “like”
Attribute           =   A tag in annotation
Value               =   “””, Value, “””
```

Figure 4.11 EBNF Grammar of the query

The previous code is called production rules and is written for describing the syntax of the complex query. The left-hand side shows the list of non-terminal symbols of the language, and the right-hand side shows the terminal as well as non-terminal symbols in a regular expression

The first line shows how this language supports many patterns for extraction; the EBNF curly brackets mean that the content of the brackets is repeatable. Note that brackets in the quotation marks are belong to the query language while others are belong to the EBNF language. For example, two patterns for extracting a sequence of one segment composed of nouns and another sequence of adjectives can be written as {[pos==“N”]} {[pos==“ADJ”]}. In this case, the algorithm will run the first pattern and add its results to the file, then run the next pattern and append it to the file. A pattern starts with a square bracket followed by the details of the node and an optional logical operator as shown in line two. It ends with an optional wild card. There are differences between the brackets in the EBNF above; for example, the bracket inside the quotation marks means that it is an actual part of the query while the one without quotation marks is for describing the query. The square bracket in EBNF indicates that an element is optional; for example, the wildcard after the node is optional and the logical operator is optional, but the match operator is not. However, the whole node can be optional. This syntax is similar to the Poliqarp syntax in [114], but it does not support the same operators as are shown in the following section.

4.3.1.1 Matching Operators

This section gives the logical operators are supported by the query as shown in Table 4.6.

Operators	**Remarks**
==	The equality operator
!=	The negation operator
re	Applies pattern matching based on regular expression
not re	Negation of regular expression
like	Another type of matching that uses % for substring searches
Within, Contains	Contextual operators
&	The logical operator of AND, used to link node conditions.
|	Another logical operator, which links nodes using OR

Table 4.6 Different types of matching operators

4.3.1.2 Wildcards

The query supports a number of wildcards, as regular expressions do. Table 4.7 outlines these notations and their functions in the annotation-based search.

Wildcard	Remarks
*	Zero or more nodes of the previous node
+	One or more nodes of the previous node
?	Zero or one of the previous node
^	The first node at the selected boundary
$	The last node at the selected boundary
(n)	Matches exactly n nodes
(n,m)	Matches n to m nodes
(n,)	Repeats the previous node from n to a n+window
(,m)	Repeats the previous node from 0 to m

Table 4.7 Wildcards operators supported by the query

4.3.2 Query Execution

Query execution is a component located between the database and the query parsing component. The main idea behind it is to manage the requests of the SQL queries that converted by the component of parsing query and get back the results as a concordance list.

```
Input: $\boldsymbol{q}$ ← a set of nodes entered by the user,
$\overline{\boldsymbol{q}}$ = $\{node_1\ node_2 \ldots node_n\}$
Output: $\overline{\boldsymbol{l}}$ is an empty array of linked lists
function FSM ($\boldsymbol{q}$)
```

```
foreach $node_i \in \boldsymbol{\bar{q}}$ do
    if $\boldsymbol{\bar{l}}$ is empty
        execute SQL query with conditions in $node_i$
            $\boldsymbol{\bar{l}}$ ← set of id for all records that satisfy conditions of $node_i$
else
    foreach $l_i \in \boldsymbol{\bar{l}}$ do
        id ← is the last element in $l_i$
        t = execute SQL query with condition in $node_i$ for record id+1
        If t is not empty
            $l_i$ ← append(t)
        else
                delete $l_i$
  return $\boldsymbol{l}$
```

Algorithm 4.1 Sequence search

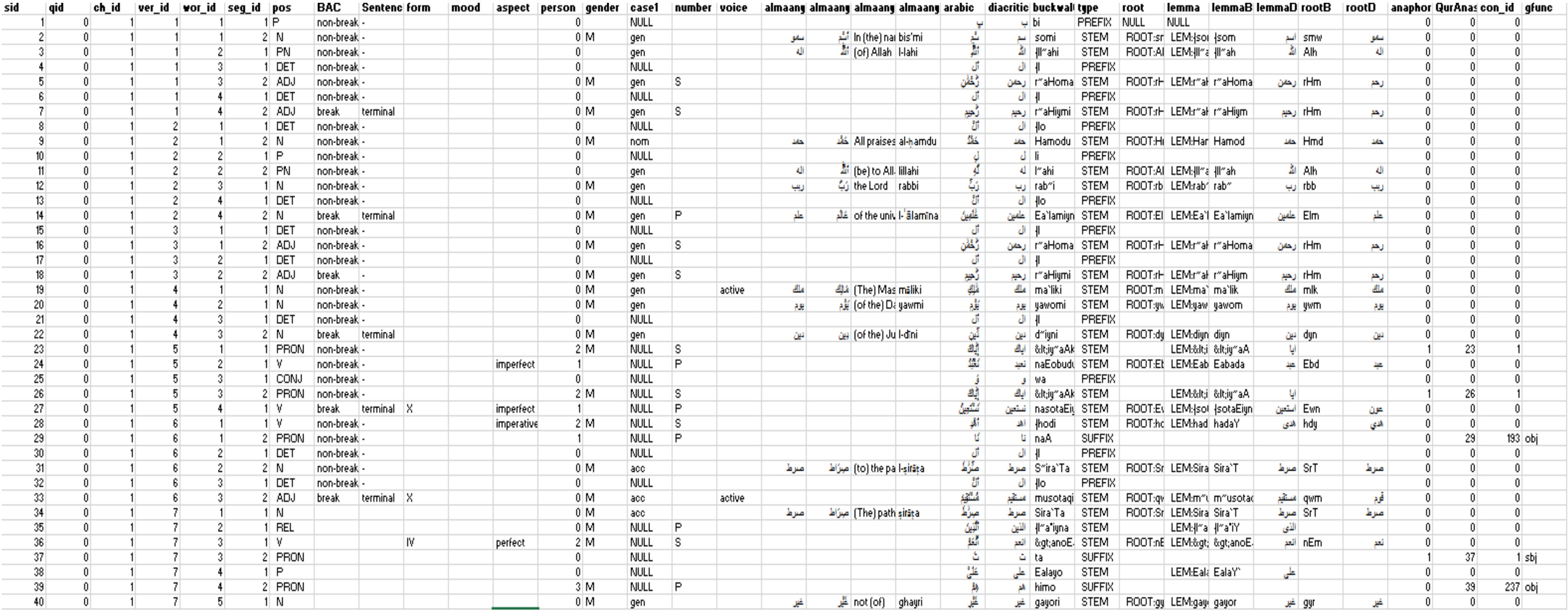
Algorithm 4.1 describes a sequence search model using the SQL query and regular expression. A parsed query is received by the FSM function in the Algorithm 4.1 as a set of nodes, and the quantifiers for the nodes are received as an array. For each node, the function finds all instances in which the given conditions are meet (the pair of attributes and values). The node also has the current position of the found instance. The second run visits the last elements for each linked list and checks whether they are null or not. If a value is null, this means that the linked list is not relevant. If it has a position number, then the second node conditions are applied for the position after the current node.

No	pos	No	pos

1	N	7	V
2	N	8	N
3	V	9	N
4	P	10	V
5	V	11	V
6	N		

Table 4.8 Sample data with one tag

Table 4.9 shows an example of the a sample of rows in a table composed of many tags. The example in Figure 4.12 shows the transition based on sample data in Table 4.8 with only one tag. Table 4.5 shows the set of attributes that can be used to make a query. The actual data is in Table 4.9 and is simplified in Table 4.8, which only has one tag and few records.



sid	qid	ch_id	ver_id	wor_id	seg_id	pos	BAC	Sentenc	form	mood	aspect	person	gender	case1	number	voice	almaany	almaany	almaany	almaany	arabic	diacritic	buckwalt	type	root	lemma	lemmaB	lemmaD	rootB	rootD	anaphor	QurAnas	con_id	gfunc
1	0	1	1	1	1	P	non-break	-				0		NULL							بِ	ب	bi	PREFIX	NULL	NULL					0	0	0	
2	0	1	1	1	2	N	non-break	-				0	M	gen			سمو	ٱسْم	In (the) nai	bis'mi	سْمِ	سم	somi	STEM	ROOT:sr	LEM:{so	{som	اسم	smw	سمو	0	0	0	
3	0	1	1	2	1	PN	non-break	-				0		gen			اله	ٱللَّه	(of) Allah	l-lahi	ٱللَّهِ	الله	{ll~ahi	STEM	ROOT:Al	LEM:{ll~a	{ll~ah	الله	Alh	اله	0	0	0	
4	0	1	1	3	1	DET	non-break	-				0		NULL							ٱل	ال	{l	PREFIX							0	0	0	
5	0	1	1	3	2	ADJ	non-break	-				0	M	gen	S						رَّحْمَٰنِ	رحمن	r~aHoma	STEM	ROOT:rH	LEM:r~ah	r~aHoma	رحمن	rHm	رحم	0	0	0	
6	0	1	1	4	1	DET	non-break	-				0		NULL							ٱل	ال	{l	PREFIX							0	0	0	
7	0	1	1	4	2	ADJ	break	terminal				0	M	gen	S						رَّحِيمِ	رحيم	r~aHiymi	STEM	ROOT:rH	LEM:r~ah	r~aHiym	رحيم	rHm	رحم	0	0	0	
8	0	1	2	1	1	DET	non-break	-				0		NULL							ٱلْ	ال	{lo	PREFIX							0	0	0	
9	0	1	2	1	2	N	non-break	-				0	M	nom			حمد	حَمْد	All praises	al-ḥamdu	حَمْدُ	حمد	Hamodu	STEM	ROOT:Hı	LEM:Ham	Hamod	حمد	Hmd	حمد	0	0	0	
10	0	1	2	2	1	P	non-break	-				0		NULL							لِ	ل	li	PREFIX							0	0	0	
11	0	1	2	2	2	PN	non-break	-				0		gen			اله	ٱللَّه	(be) to All	lillahi	لَّهِ	له	l~ahi	STEM	ROOT:Al	LEM:{ll~a	{ll~ah	الله	Alh	اله	0	0	0	
12	0	1	2	3	1	N	non-break	-				0	M	gen			ربب	رَبّ	the Lord	rabbi	رَبِّ	رب	rab~i	STEM	ROOT:rb	LEM:rab~	rab~	رب	rbb	ربب	0	0	0	
13	0	1	2	4	1	DET	non-break	-				0		NULL							ٱلْ	ال	{lo	PREFIX							0	0	0	
14	0	1	2	4	2	N	break	terminal				0	M	gen	P		علم	عَالَم	of the univ	l-ʿālamīna	عَٰلَمِينَ	علمين	Ea`lamiyn	STEM	ROOT:El	LEM:Ea`l	Ea`lamiyn	علمين	Elm	علم	0	0	0	
15	0	1	3	1	1	DET	non-break	-				0		NULL							ٱل	ال	{l	PREFIX							0	0	0	
16	0	1	3	1	2	ADJ	non-break	-				0	M	gen	S						رَّحْمَٰنِ	رحمن	r~aHoma	STEM	ROOT:rH	LEM:r~ah	r~aHoma	رحمن	rHm	رحم	0	0	0	
17	0	1	3	2	1	DET	non-break	-				0		NULL							ٱل	ال	{l	PREFIX							0	0	0	
18	0	1	3	2	2	ADJ	break	-				0	M	gen	S						رَّحِيمِ	رحيم	r~aHiymi	STEM	ROOT:rH	LEM:r~ah	r~aHiym	رحيم	rHm	رحم	0	0	0	
19	0	1	4	1	1	N	non-break	-				0	M	gen		active	ملك	مَالِك	(The) Mas	māliki	مَٰلِكِ	ملك	ma`liki	STEM	ROOT:m	LEM:ma`	ma`lik	ملك	mlk	ملك	0	0	0	
20	0	1	4	2	1	N	non-break	-				0	M	gen			يوم	يَوْم	(of the) Da	yawmi	يَوْمِ	يوم	yawomi	STEM	ROOT:yw	LEM:yaw	yawom	يوم	ywm	يوم	0	0	0	
21	0	1	4	3	1	DET	non-break	-				0		NULL							ٱل	ال	{l	PREFIX							0	0	0	
22	0	1	4	3	2	N	break	terminal				0	M	gen			دين	دِين	(of the) Ju	l-dīni	دِّينِ	دين	d~iyni	STEM	ROOT:dy	LEM:diyn	diyn	دين	dyn	دين	0	0	0	
23	0	1	5	1	1	PRON	non-break	-				2	M	NULL	S						إِيَّاكَ	اياك	<iy~aAk	STEM		LEM:<i	<iy~aA	ايا			1	23	1	
24	0	1	5	2	1	V	non-break	-			imperfect	1		NULL	P						نَعْبُدُ	نعبد	naEobudu	STEM	ROOT:Eb	LEM:Eab	Eabada	عبد	Ebd	عبد	0	0	0	
25	0	1	5	3	1	CONJ	non-break	-				0		NULL							وَ	و	wa	PREFIX							0	0	0	
26	0	1	5	3	2	PRON	non-break	-				2	M	NULL	S						إِيَّاكَ	اياك	<iy~aAk	STEM		LEM:<i	<iy~aA	ايا			1	26	1	
27	0	1	5	4	1	V	break	terminal	X		imperfect	1		NULL	P						نَسْتَعِينُ	نستعين	nasotaEiy	STEM	ROOT:Ew	LEM:{so	{sotaEiyn	استعين	Ewn	عون	0	0	0	
28	0	1	6	1	1	V	non-break	-			imperative	2	M	NULL	S						ٱهْدِ	اهد	{hodi	STEM	ROOT:hd	LEM:had	hadaY	هدى	hdy	هدي	0	0	0	
29	0	1	6	1	2	PRON	non-break	-				1		NULL	P						نَا	نا	naA	SUFFIX							0	29	193	obj
30	0	1	6	2	1	DET	non-break	-				0		NULL							ٱل	ال	{l	PREFIX							0	0	0	
31	0	1	6	2	2	N	non-break	-				0	M	acc			صرط	صِرَاط	(to) the pa	l-ṣirāṭa	صِّرَٰطَ	صرط	S~ira`Ta	STEM	ROOT:Sr	LEM:Sira	Sira`T	صرط	SrT	صرط	0	0	0	
32	0	1	6	3	1	DET	non-break	-				0		NULL							ٱلْ	ال	{lo	PREFIX							0	0	0	
33	0	1	6	3	2	ADJ	break	terminal	X			0	M	acc		active					مُسْتَقِيمَ	مستقيم	musotaqi	STEM	ROOT:qw	LEM:m~u	m~usotaq	مستقيم	qwm	قوم	0	0	0	
34	0	1	7	1	1	N						0	M	acc			صرط	صِرَاط	(The) path	ṣirāṭa	صِرَٰطَ	صرط	Sira`Ta	STEM	ROOT:Sr	LEM:Sira	Sira`T	صرط	SrT	صرط	0	0	0	
35	0	1	7	2	1	REL						0	M	NULL	P						ٱلَّذِينَ	الذين	{l~a*iyna	STEM		LEM:{l~a	{l~a*iY	الذى			0	0	0	
36	0	1	7	3	1	V			IV		perfect	2	M	NULL	S						أَنْعَمْ	انعم	>anoE	STEM	ROOT:nE	LEM:>	>anoE	انعم	nEm	نعم	0	0	0	
37	0	1	7	3	2	PRON						0		NULL							تَ	ت	ta	SUFFIX							1	37	1	sbj
38	0	1	7	4	1	P						0		NULL							عَلَيْ	على	Ealayo	STEM		LEM:Eala	EalaY`	على			0	0	0	
39	0	1	7	4	2	PRON						3	M	NULL	P						هِمْ	هم	himo	SUFFIX							0	39	237	obj
40	0	1	7	5	1	N						0	M	gen			غير	غَيْر	not (of)	ghayri	غَيْرِ	غير	gayori	STEM	ROOT:gy	LEM:gay	gayor	غير	gyr	غير	0	0	0	

Table 4.9 Labels and their values from the first 40 rows in a morphological table

{[pos==“N”] [pos==“N”][pos==“V”](2)}

Runs	Conditions	Quantifier	Output
1	[pos==“N”]	1	1 → 1 2 → 2 3 → 6 4 → 8 5 → 9
2	[pos==“N”]	1	1 → 1 2 2 → 2 N 3 → 6 N 4 → 8 9 5 → 8 N
3	[pos=“V”]	2	1 → 1 2 3 2 → 2 N 3 → 6 N 4 → 8 9 10 5 → 8 N 1 → 1 2 3 N 2 → 2 N 3 → 6 N 4 → 8 9 10 11 5 → 8 N

Figure 4.12 An implementation for a annotation-based search

Figure 4.12 shows an example of how the algorithm runs the annotation-based search over several resources. This example based on the sample data annotated with one tag for explanation purposes which can be seen in Table 4.8. The first column indicates the nodes of the query. In this example, we have three nodes ([pos==“N”] [pos==“N”] [pos==“V”]). Each run adds the instances that satisfy the condition of the current node.

The second column shows the current node and the conditions that must be met in order to be added. For instance, the condition of the first node is any word that tagged with "Noun" as Part of Speech (POS), while the condition of the third node is that any word tagged with "Verb". The third column represents the repetition information of the node. As shown in the third row, there are two repetitions because this node was followed by the quantifier symbol (n), which is explained in Table 4.7. The third column shows an array of linked lists, which represent the retrieved instances. The size of list is based on the condition of the first node in the query. In our example, there are 5 records are tagged with "N", therefore, all linked list will have one of the ids of these records as the first element. For the second node, the search will be limited to the words that are following the current one in the linked list. For example, the first linked list have the id "1" as the first element. In order to search of the second element we only need to check the word located in 1+1 which is the following one. If this word is tagged with the same condition stated in the second node then the id of the founded word will be added. Otherwise null will be an indication the word is not founded. Every node generates an addition to this array depending on the given conditions and the linked lists, and no null values will be considered as the relevant value retrieved at the final step. As a result of this example, there is only one instance that satisfies the given query for this example, segments 8 to 11 in Table 4.8.

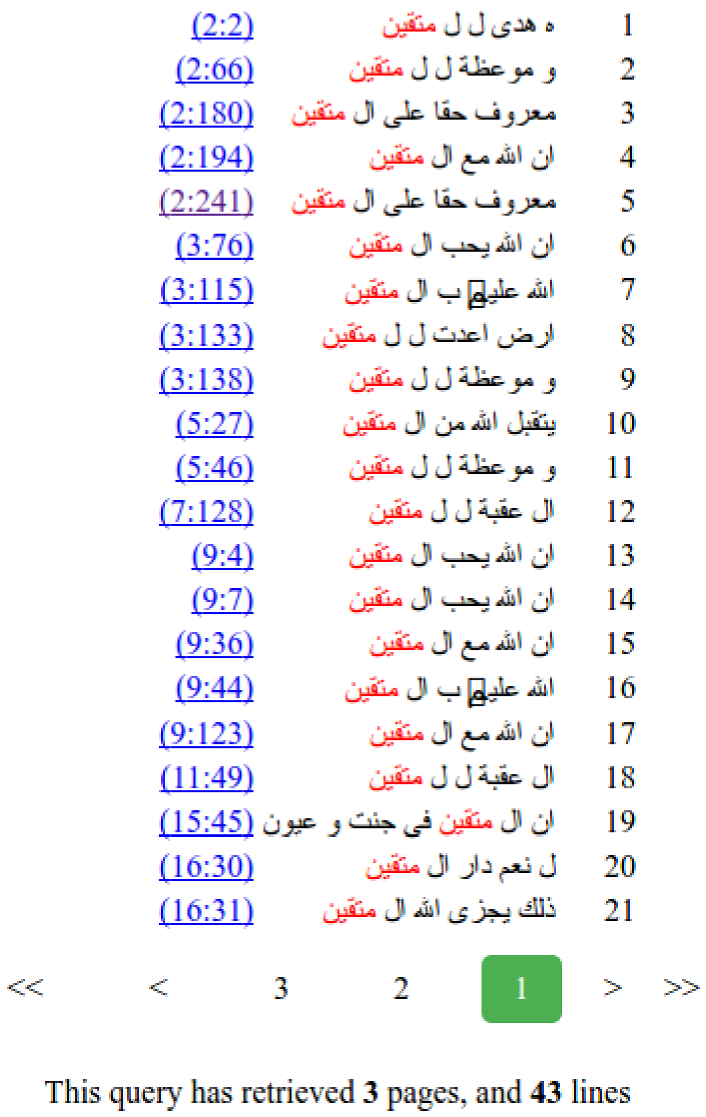
4.4 Experiment of Extraction

In this section, the usage of AQD and the annotation-based search will be described and evaluated by different examples of linguistic phenomena occurring in the Arabic text, starting with a simple search and ending with a very complex extraction task.

4.4.1 Simple Annotation Based Search

Each annotation that was included in the AQD can be used as an attribute, as is shown in the following query. The simple query is composed of one node and does not have any attributes or conditions, such as {[]}. This query will retrieve every segment stored in the database. Another simple example can be shown in the following, which extracts all occurrences of the word ‘متقين.’ This query output is shown in Figure 4.13. The attribute *diacritics* represents the diacriticised Qur’anic text. Several different features can be used as attributes, and are shown in Table 4.5.

{[diacritics==”متقين”]}



1	ه هدى ل ل متقين	(2:2)
2	و موعظة ل ل متقين	(2:66)
3	معروف حقا على ال متقين	(2:180)
4	ان الله مع ال متقين	(2:194)
5	معروف حقا على ال متقين	(2:241)
6	ان الله يحب ال متقين	(3:76)
7	الله عليم ب ال متقين	(3:115)
8	ارض اعدت ل ل متقين	(3:133)
9	و موعظة ل ل متقين	(3:138)
10	يتقبل الله من ال متقين	(5:27)
11	و موعظة ل ل متقين	(5:46)
12	ال عقبة ل ل متقين	(7:128)
13	ان الله يحب ال متقين	(9:4)
14	ان الله يحب ال متقين	(9:7)
15	ان الله مع ال متقين	(9:36)
16	الله عليم ب ال متقين	(9:44)
17	ان الله مع ال متقين	(9:123)
18	ال عقبة ل ل متقين	(11:49)
19	ان ال متقين في جنت و عيون	(15:45)
20	ل نعم دار ال متقين	(16:30)
21	ذلك يجزى الله ال متقين	(16:31)

<< < 3 2 1 > >>

This query has retrieved **3** pages, and **43** lines

Figure 4.13 The output of a simple query

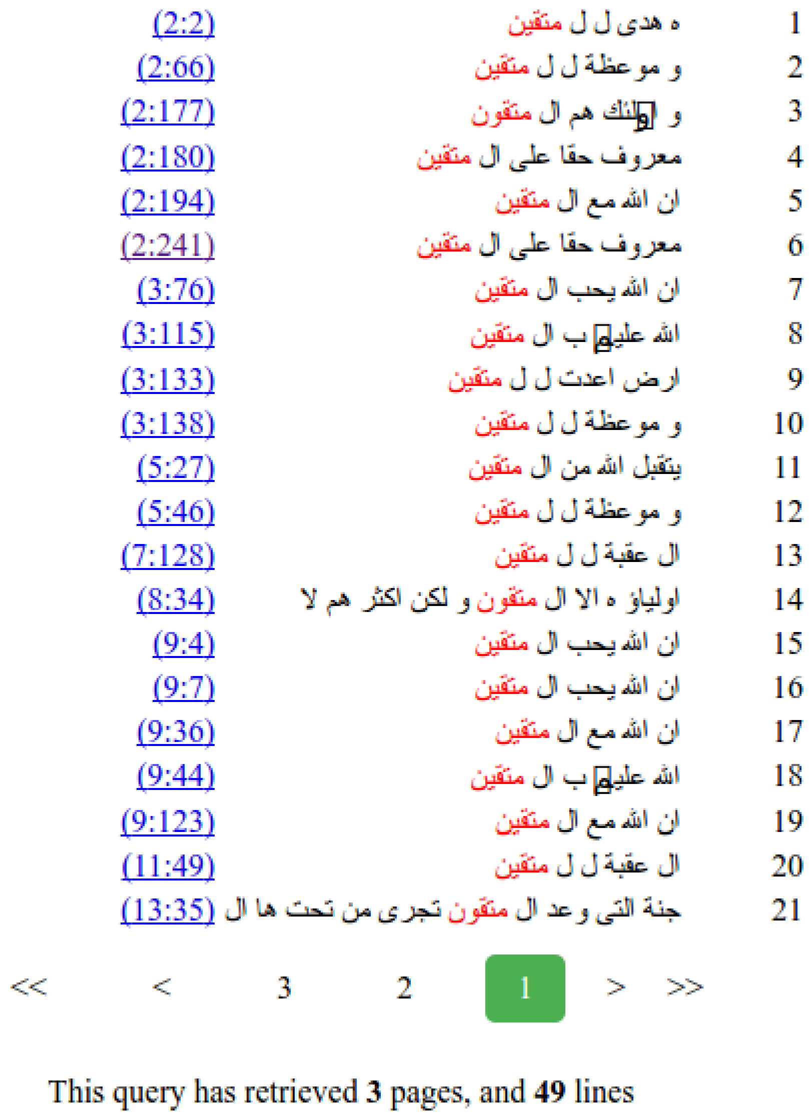
The retrieved instances are matched to the given word. This will only search for one node matched by the given text. If there are other words with the same meaning that we wish to extract, this query will not be able to retrieve them.

Searching over lemma instead of the exact text of words will extract all occurrences of a word that belongs to the given lemma. The example shown in Listing 4.1 retrieves more instances because it is based on the lemma and not the word.

Listing 4.1

```
{[LemmaD=="متقين"]}
```

This query will retrieve all words that belong to the given lemma. In this query, the output will be larger because words like 'متقون' will be considered as well.



1	ه هدى ل ل متقين	(2:2)
2	و موعظة ل ل متقين	(2:66)
3	و ا□لئك هم ال متقون	(2:177)
4	معروف حقا على ال متقين	(2:180)
5	ان الله مع ال متقين	(2:194)
6	معروف حقا على ال متقين	(2:241)
7	ان الله يحب ال متقين	(3:76)
8	الله عليم□ ب ال متقين	(3:115)
9	ارض اعدت ل ل متقين	(3:133)
10	و موعظة ل ل متقين	(3:138)
11	يتقبل الله من ال متقين	(5:27)
12	و موعظة ل ل متقين	(5:46)
13	ال عقبة ل ل متقين	(7:128)
14	اولياؤ ه الا ال متقون و لكن اكثر هم لا	(8:34)
15	ان الله يحب ال متقين	(9:4)
16	ان الله يحب ال متقين	(9:7)
17	ان الله مع ال متقين	(9:36)
18	الله عليم□ ب ال متقين	(9:44)
19	ان الله مع ال متقين	(9:123)
20	ال عقبة ل ل متقين	(11:49)
21	جنة التى وعد ال متقون تجرى من تحت ها ال	(13:35)

This query has retrieved **3** pages, and **49** lines

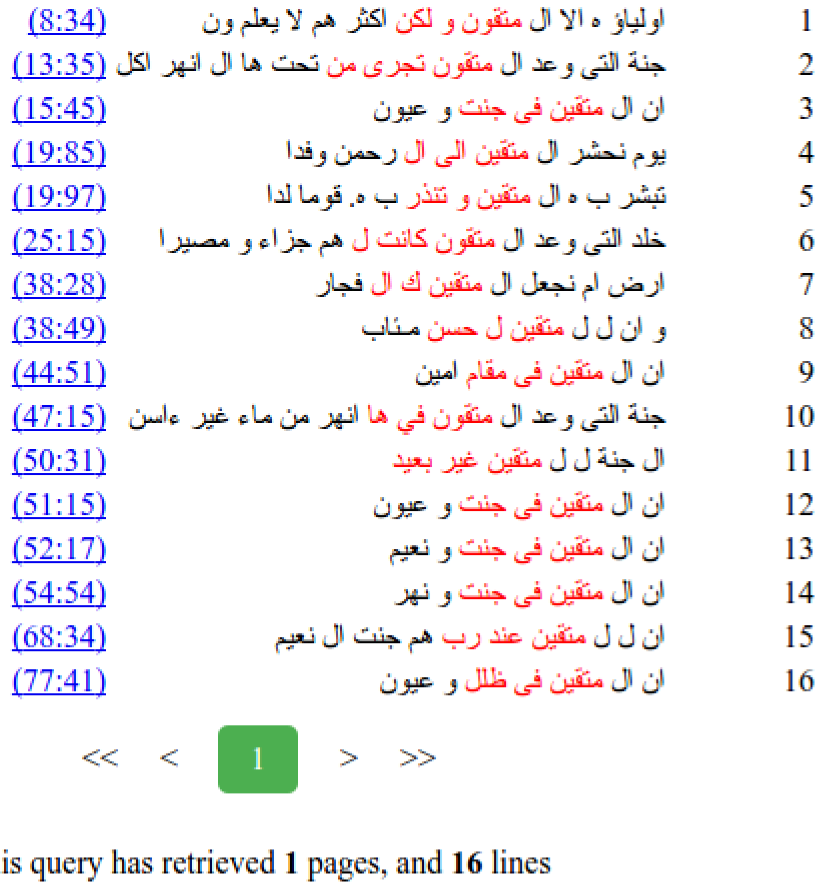
Figure 4.14 Search by lemma

This tool also supports all morphological annotations produced by [109]. To measure the quality of the extracted results, the QAC morphological search was used, and the results are shown in Figure 4.14.

Another feature is the blank node, which lets the user add as many as needed n-grams. The blank node just retrieves the next segment without checking content.

Listing 4.2 {[LemmaD==”متقين”][][]}

This is an example of how to use blank nodes. Two blank nodes were added to retrieve the two segments, followed the given lemma. Figure 4.15 shows the output of this query.



(8:34)	اولياؤ ه الا ال متقون و لكن اكثر هم لا يعلم ون	1
(13:35)	جنة التى وعد ال متقون تجرى من تحت ها ال انهر اكل	2
(15:45)	ان ال متقين فى جنت و عيون	3
(19:85)	يوم نحشر ال متقين الى ال رحمن وفدا	4
(19:97)	تبشر ب ه ال متقين و تنذر ب ه. قوما لدا	5
(25:15)	خلد التى وعد ال متقون كانت ل هم جزاء و مصيرا	6
(38:28)	ارض ام نجعل ال متقين ك ال فجار	7
(38:49)	و ان ل ل متقين ل حسن مـئاب	8
(44:51)	ان ال متقين فى مقام امين	9
(47:15)	جنة التى وعد ال متقون في ها انهر من ماء غير ءاسن	10
(50:31)	ال جنة ل ل متقين غير بعيد	11
(51:15)	ان ال متقين فى جنت و عيون	12
(52:17)	ان ال متقين فى جنت و نعيم	13
(54:54)	ان ال متقين فى جنت و نهر	14
(68:34)	ان ل ل متقين عند رب هم جنت ال نعيم	15
(77:41)	ان ال متقين فى ظلل و عيون	16

This query has retrieved **1** pages, and **16** lines

Figure 4.15 Add two node as a search window

The query also supports regular expression inside the values of the given attribute. Retrieval is accomplished by providing part of the text using the “re” operator, as seen in Listing 4.3.

Listing 4.3 {[diacritics re “%متقين”]}

4.4.2 Construct State (iDAfa)

iDAfa is a construction of two nouns, or what is known in English as a compound or possessive construction. iDAfa is important for dependency parsing and is also known as an annexation construct. It makes up more than 75% of Arabic sentences.

The following example in Listing 4.4 shows two things. First, it shows how to enables multiple rules for a single task. Second, it shows how to extract the construct of an iDAfa. The following example shows three rules that were defined to extract iDAfa.

To prepare rules for extraction, an expert of Arabic language must define them. We used the same rules applied in [115]. A construct-state in Arabic is composed of two parts, a noun and either a definite noun or a pronoun in genitive case. We found 4,348 iDAfa constructs in the Qur’an.

Listing 4.4

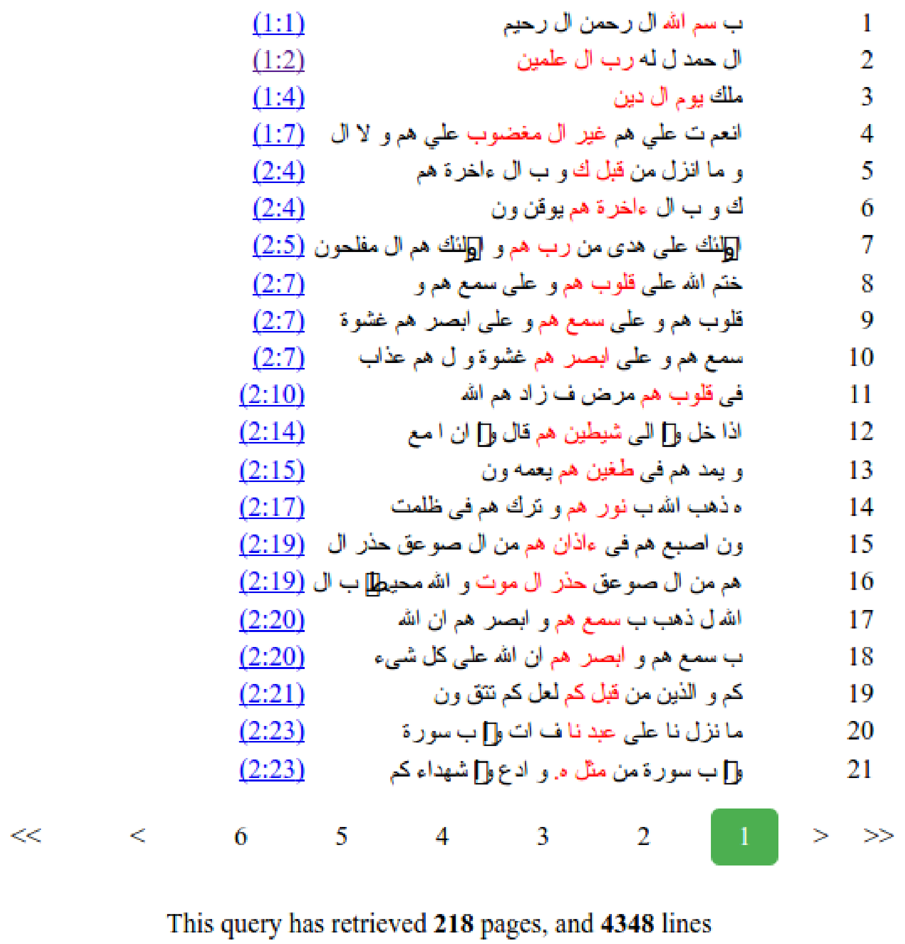
```
{[pos=="N" & case=="gen"][pos=="PRON"]}
{[ pos=="N"][pos == "PN" & case=="gen"]}
{[ pos=="N"][pos == "DET"][pos == "N" & case=="gen"]}
```

The same task can be represented using context-free grammar, as in Listing 4.5.

Listing 4.5

```
GDN->{[pos == "PN" & case=="gen"]}
{[pos=”DET”][pos=="N" & case=="gen"]}
GIN->{[pos=="N" & case=="gen"]
IN->{[ pos=="N"]}
iDAfa ->{[GIN] [pos=="PRON"]} {[IN][GDN]} {[IN][GDN]}
```

This representation allows users to employ their defined tags as nonterminal, such as GIN or IN in Listing 4.5.



1	ب سم الله ال رحمن ال رحيم	(1:1)
2	ال حمد ل له رب ال علمين	(1:2)
3	ملك يوم ال دين	(1:4)
4	انعم ت علي هم غير ال مغضوب علي هم و لا ال	(1:7)
5	و ما انزل من قبل ك و ب ال ءاخرة هم	(2:4)
6	ك و ب ال ءاخرة هم يوقن ون	(2:4)
7	□لئك على هدى من رب هم و □لئك هم ال مفلحون	(2:5)
8	ختم الله على قلوب هم و على سمع هم و	(2:7)
9	قلوب هم و على سمع هم و على ابصر هم غشوة	(2:7)
10	سمع هم و على ابصر هم غشوة و ل هم عذاب	(2:7)
11	فى قلوب هم مرض ف زاد هم الله	(2:10)
12	اذا خل □ الى شيطين هم قال □ ان ا مع	(2:14)
13	و يمد هم فى طغين هم يعمه ون	(2:15)
14	ه ذهب الله ب نور هم و ترك هم فى ظلمت	(2:17)
15	ون اصبع هم فى ءاذان هم من ال صوعق حذر ال	(2:19)
16	هم من ال صوعق حذر ال موت و الله محيط□ ب ال	(2:19)
17	الله ل ذهب ب سمع هم و ابصر هم ان الله	(2:20)
18	ب سمع هم و ابصر هم ان الله على كل شىء	(2:20)
19	كم و الذين من قبل كم لعل كم تتق ون	(2:21)
20	ما نزل نا على عبد نا ف ات □ ب سورة	(2:23)
21	□ ب سورة من مثل ه. و ادع □ شهداء كم	(2:23)

This query has retrieved **218** pages, and **4348** lines

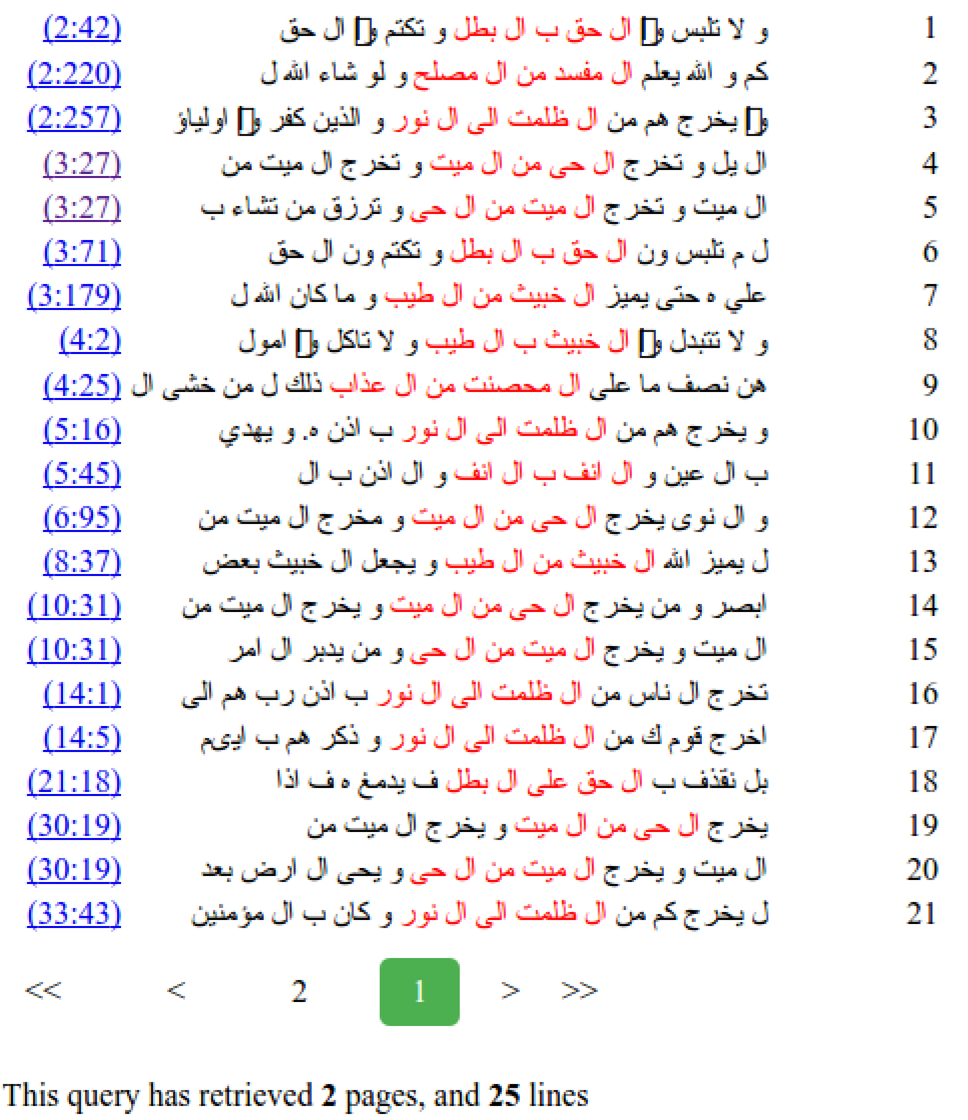
Figure 4.16 The grammatical structure (iDAfa) based on three patterns

4.4.3 Ontological relations between Definite Nouns

This environment could help in extracting complex relationships, such as anatomy, in the next example. In this experiment, the focus was only on the antonyms in prepositional phrases of the Qur'an. The following script in Listing 4.6 is the query used to extract the triples in Figure 4.17.

Listing 4.6

```
{[trim(pos)=="DET"][trim(pos)=="N" & trim(case1)=="gen" &
trim(gender)=="F" &
trim(num)=="P"][trim(pos)=="P"][trim(pos)=="DET"][trim(pos)=="N" &
trim(case1)=="gen" & trim(gender)=="M" & trim(num)==""]}
{[trim(pos)=="DET"][trim(pos)=="N" & trim(case1)=="acc" &
trim(gender)=="M" &
trim(num)=="S"][trim(pos)=="P"][trim(pos)=="DET"][trim(pos)=="N" &
trim(case1)=="gen" & trim(gender)=="M" & trim(num)=="S"]}
{[trim(pos)=="DET"][trim(pos)=="N" & trim(case1)=="acc" &
trim(gender)=="M" &
trim(num)==""][trim(pos)=="P"][trim(pos)=="DET"][trim(pos)=="N" &
trim(case1)=="gen" & trim(gender)=="M" & trim(num)=="" &
trim(voice)=="active"]}
{[trim(pos)=="DET"][trim(pos)=="N" & trim(case1)=="gen" &
trim(gender)=="F" &
trim(num)=="P"][trim(pos)=="P"][trim(pos)=="DET"][trim(pos)=="N" &
trim(case1)=="gen" & trim(gender)=="M" & trim(num)=="P"]}
{[trim(pos)=="DET"][trim(pos)=="N" & trim(case1)=="gen" &
trim(gender)=="M" &
trim(num)==""][trim(pos)=="P"][trim(pos)=="DET"][trim(pos)=="N" &
trim(case1)=="gen" & trim(gender)=="M" & trim(num)=="" &
trim(voice)=="active"]}
```



(2:42)	و لا تلبس و□ ال حق ب ال بطل و تكتم و□ ال حق	1
(2:220)	كم و الله يعلم ال مفسد من ال مصلح و لو شاء الله ل	2
(2:257)	و□ يخرج هم من ال ظلمت الى ال نور و الذين كفر و□ اولياؤ	3
(3:27)	ال يل و تخرج ال حى من ال ميت و تخرج ال ميت من	4
(3:27)	ال ميت و تخرج ال ميت من ال حى و ترزق من تشاء ب	5
(3:71)	ل م تلبس ون ال حق ب ال بطل و تكتم ون ال حق	6
(3:179)	علي ه حتى يميز ال خبيث من ال طيب و ما كان الله ل	7
(4:2)	و لا تتبدل و□ ال خبيث ب ال طيب و لا تاكل و□ امول	8
(4:25)	هن نصف ما على ال محصنت من ال عذاب ذلك ل من خشى ال	9
(5:16)	و يخرج هم من ال ظلمت الى ال نور ب اذن ه. و يهدي	10
(5:45)	ب ال عين و ال انف ب ال انف و ال اذن ب ال	11
(6:95)	و ال نوى يخرج ال حى من ال ميت و مخرج ال ميت من	12
(8:37)	ل يميز الله ال خبيث من ال طيب و يجعل ال خبيث بعض	13
(10:31)	ابصر و من يخرج ال حى من ال ميت و يخرج ال ميت من	14
(10:31)	ال ميت و يخرج ال ميت من ال حى و من يدبر ال امر	15
(14:1)	تخرج ال ناس من ال ظلمت الى ال نور ب اذن رب هم الى	16
(14:5)	اخرج قوم ك من ال ظلمت الى ال نور و ذكر هم ب ايىم	17
(21:18)	بل نقذف ب ال حق على ال بطل ف يدمغ ه ف اذا	18
(30:19)	يخرج ال حى من ال ميت و يخرج ال ميت من	19
(30:19)	ال ميت و يخرج ال ميت من ال حى و يحى ال ارض بعد	20
(33:43)	ل يخرج كم من ال ظلمت الى ال نور و كان ب ال مؤمنين	21

<< < 2 1 > >>

This query has retrieved **2** pages, and **25** lines

Figure 4.17 Some of the antonymic relations in prepositional phrases

Figure 4.17 shows the ontological relationships between Qur'anic nouns in prepositional phrases. These type of relations need very complex rules that combines several features to be included in the query to extract these instances (Table 4.5).

4.5 Summary

In this chapter, a new database called AQD for available annotations of the Qur’an has been combined from different annotation sources. AQD includes morphological, chronological, ontological and structural annotations. The AQD reveals that the number of ontological relations found in the combined resources is far smaller in comparisons to other types of relations. This chapter also conducted an implementation of an annotation-based search for different types of annotations for the Qur’an. The query supports differently-resourced annotations. This tool is very important, as it allows Qur’anic researchers to mine and discover new patterns within these annotations. The tool used in the experiments is available on our website, together with the source code of the annotation-based search[1]. The package of the source code and the database are available for download. The experimental results show that the provided environment can extract a syntactic construct of Arabic called iDAfa, and antonym relationships occur in prepositional phrases.

This work supports more annotations for the annotation-based search of the Qur’an and allows users to make a complex query that integrates a number of annotations in one single query. In addition, it supports context-free grammar for complex extraction tasks like those shown in the extraction of iDAfa.

There is much room for further work here. We plan to investigate the possibility of adding more annotations and features for the AQD.

[1] Salrehaili.com/AQD

Part III

Modelling Qur’anic Ontology Learning

Chapter 5

Concept Identification

Understanding a domain requires having enough knowledge about its relevant terms, concepts and relationships. Concepts are often listed with articles as keywords or key terms to provide a quick understanding about the content. The identification of domain concepts is a crucial step in many natural language processing applications. Concept extraction from text relies on domain-specific terms extraction methods because terms are the correspond to linguistic representation of concepts [116].

The term extraction is a process of obtaining a set of terms that represent a domain of a given text. Only few researchers have paid attention to the area of Automatic Concept Identification from Arabic text of the Qur'an. Moreover, Using Arabic text instead of a translation may results in more concepts extraction. Furthermore, Concepts extraction is requires as a first step for many NLP applications and this may increase the impact of my research. The majority of term extraction research projects conducted for the Qur'an have used translated text instead of the original Classical Arabic text of the Qur'an. The extraction of terms from the original Arabic text rather than a translation may help in retrieving more relevant terms, due to the lack of Islamic equivalents of some Qur'an terms in other languages. A number of evidence and examples discussed for non-equivalence in the process of translating from Arabic to English can be found in [14].

This chapter demonstrates an unsupervised method for the acquisition of a list of domain-specific terms from the Arabic text of the

Qur’an. The produced list of terms was validated based on three types of evaluation metrics. First, an existing datasets. Second, partially validation for a certain parts of the Qur’an. Third, a common evaluation metric for ranked list called Average Precision.

This chapter is also provides the method of extraction as a pipelined architecture for concept identification and three types of evaluation metrics with a number of attempts to improve the Arabic term extraction. To the best of my knowledge, this work is the first in Arabic that uses lexical and inflectional information for terms candidate extraction. In addition, it uses different schema from previous work for ranking these terms, which computes a combination of domain-specific and statistical for weighting and extracting the domain-specific terms. In addition, it provides a resource called AQT, which contains of 10351 instances of manually validated terms to be used in evaluation and as a training for supervised concept identification systems. This resource used as a training for machine learning experiment on domain specific extraction in **Error! Reference source not found.**. This chapter is based on our published paper [117]. The last part of this chapter presents an experiment of applying machine learning technique conducted for learning Qur’anic domain-specific terms. We exploit the resource of AQT which was manually validated to build a model using a machine learning algorithm. The result is compared to the previous results.

5.1 Concept Identification

In order to identify a list of domain concepts, most of the available methods composed of three prerequisite steps. (1) Concept-like extraction, (2) relevance and (3) ranking.

Firstly, linguistic method are applied for candidate selection which is a step in which concept-like items are extracted. There are two methods for extracting concept candidates in ontology learning systems. The first method is to extract single-word terms, then collocation are generated in order to derive multi-word terms such as in [76]. However as [118] addressed that this method will extract more single-word terms as concepts which means the learned ontology will missed many important multi-word terms. Multi-word terms carry more meaning than the single-word terms [37], [39]. Approximately 85% of domain-specific concepts are made of MWT [38]. In the second method, noun phrases are extracted which then generate all possible combinations of terms that belong to the extracted noun phrases. We follow the second method for extracting the concept candidates because it generates more multi-word terms than single-word terms.

Secondly, statistical methods are applied to generated candidates to calculate their important and relevance to the domain. Because some of generated candidates are not necessary to represent the domain. In this step We follow a scheme combined from statistical and domain-specific knowledge information for calculating the importance of the candidates. According to [71], domain-specific knowledge resources should be used to help term extraction methods.

Thirdly, ranking the weighted candidates according to the assigned weight using the scheme applied in the previous step and pick the most important one from the top.

5.1.1 Discovering Concept-candidates in The Qur’an Text

The aim of this task is to identify the possible candidates of terms for the Arabic text of the Qur’an. Most of available works rely on traditional patterns that used in other languages while patterns of this work is based on manual annotation for a selected chapter of the Qur’an. In other words we are looking for syntactic patterns that a sequence of words should fulfil in order to be considered as a term candidate.

Firstly, related research were reviewed particularly, that conducted the concept extraction for the Qur’an using Arabic text. For example, [76] used Noun, Noun-Adjective, Noun-Noun, and Verb-Noun for candidate selecting. [7] applied KP-Miner which is described in section 4.3.4. [77] only use unigram as a candidate.

Secondly, a manual annotation was conducted for one chapter of the Qur’an, three annotators who are native speakers and have an experience in semantic annotation as well as memorised some chapters of the Qur'an have manually extracted all patterns for concepts and terms from chapter number 29 of the Qur'an. One of these annotators is the author of this thesis. Approximately 470 terms with their syntactic patterns were extracted. This task took a long time for annotating each verse with its list of terms and patterns and after that review the annotation, although we are already know that this type of tasks is time and effort consuming. In addition, only few researchers were interested to involve in such a work. During the annotation, the annotator read all the targeted chapter and for every verse he is asked to extract candidates based on the interpretation information from two common books of Tafsir; Ibn Katheer [119] and Almuyasser. Tafsir books have comments on each verse of the Qur'an, highlighting the hidden information about the verse and its related events, characteristic and the reason of their revelation.

Three previous works such as [41], [51], [120] also is considered as they presented some of Qur'anic topics and pronouns references. Table 5.1 shows the first 15 lines of the manual annotation of the chapter number 29. Column "c" indicates the chapter number that the terms are mentioned in, column "v" is the verse number, and column "t" is the token, column "s" is the start and the end of segments. POS represents the patterns of the part of terms as a sequences of POS tag values. The next column represents the version of Arabic text without diacritical marks. Following by the English translation of the terms and the occurrences. Semantic tag refers to the category these terms belong to . Finally, Taffsir taf which indicates the meaning category according to the taffsir book we based on.

No	c	v	t	s	POS pattern	Arabic text	English text	occurrence	Semantic Tag	taffsir tag
1	29	2	2	1-2	DET.N	الناس	the people	5	people, believers	
2	29	3	3-5	1-1, 1-1, 1-2	REL P N.PRON	الذين من قبلهم	Those who were before them	1	people, believers	ancestors, believers,
3	29	3	7	1-1	PN	الله	Allah	41	Allah	
4	29	3	8-9	1-1, 1-2	REL V.PRON	الذين صدقوا	Those who are the truthful	1	people, truthful	believers,
5	29	3	11	1-2	DET.N	الكاذبين	the liars	1	people, liars	disbelievers,
6	29	4	3-5	1-1, 1-2, 1-2	REL V.PRON DET.N	الذين يعملون السيئات	Those who do evil deeds	1	people, mušrikūn	disbelivers,
7	29	4	5	1-2	DET.N	السيئات	evil deeds	1	act, bad deeds, polytheism	
8	29	4	9-10	1-1, 1-2	REL V.PRON	ما يحكمون	What they judge	1	act, bad deeds, polytheism	
9	29	5	1-5	1-1, 1-1, 1-1, 1-1, 1-1	REL V V N PN	من كان يرجوا لقاء الله	Whoever is hopes meeting Allah	1	people, believers	
10	29	5	4	1-1	N	لقاء	meeting	2	result, rewards	
11	29	5	5	1-1	PN	الله	Allah	41	allah	
12	29	5	4-5	1-1, 1-1	N PN	لقاء الله	the meeting with Allah	1	result, rewards	death, hereafter,
13	29	5	7	1-1	N	أجل	Term	2	result, rewards	
14	29	5	8	1-1	PN	الله	Allah	41	Allah	
15	29	5	7-8	1-1, 1-1	N PN	أجل الله	Ther term of Allah	1	result, rewards	death, hereafter,

Table 5.1 First 15 concept-like from chapter no 29 of the Qur'an.

No	Syntactic Patterns	Occurrences
1	N	25136
2	DET.N	7488
3	PN	3911
4	REL V	2919
5	N N	2621
6	ADJ	1961
7	REL V.PRON	1886
8	N V.PRON	1377
9	N DET.N	1328
10	N ADJ	1049

Table 5.2 The top 10 most frequently occurring concept-like patterns in the Qur'an.

Table 5.2 shows the most 10 frequent syntactic patterns based on the manual extraction above for the entire Qur'an with their occurrences. All syntactic patterns can be seen in Table B.1 in Appendix B .

5.1.2 Modular Design

As a result of the Qur'anic concept-like discovery task, which described in Section 5.1.1, we design a new framework, shown in Figure 5.1, for concept identification combining methods from linguistic, statistics and domain-specific knowledge resources.

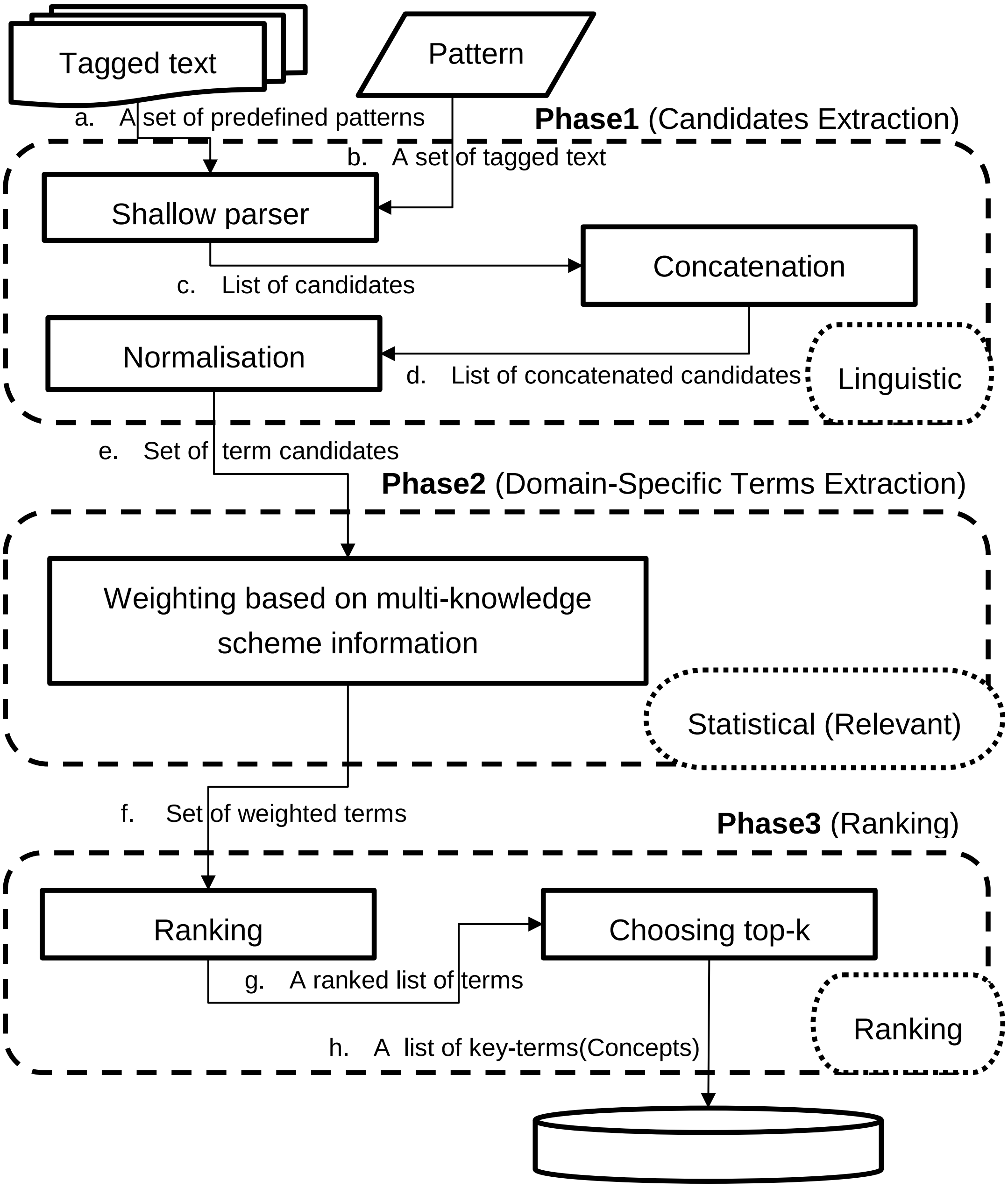

Tagged text
Pattern
a. A set of predefined patterns
Phase1 (Candidates Extraction)
b. A set of tagged text
Shallow parser
Concatenation
c. List of candidates
Normalisation
d. List of concatenated candidates
Linguistic
e. Set of term candidates
Phase2 (Domain-Specific Terms Extraction)
Weighting based on multi-knowledge scheme information
Statistical (Relevant)
f. Set of weighted terms
Phase3 (Ranking)
Ranking
Choosing top-k
g. A ranked list of terms
Ranking
h. A list of key-terms(Concepts)

Figure 5.1 The architecture of Concept Learning.

This pipelined architecture of three phases was proposed for extracting Qur'anic concepts. Figure 5.1 depicts each phase in a rounded dashed rectangle with their title in the top right. First phase, receives a tagged text with POS and set of patterns to extract a set of term candidates using linguistic method. The next phase based on a statistical method, which receives the candidates and compute the importance and relevance measurements to assign them to the term as a weight. The final phase, takes the weighted candidates and rank them according to certain number from the top of the ranked list.

5.1.2.1 Candidates Extraction

This is the first step in term extraction and known as “Unithood” in [36], [121], [122], which aims to extract the possible sequences of words that form a stable lexical unit.

5.1.2.1.1 Parsing

In this phase, a shallow parsing technique from NLTK and based on examples explained in [123], [124] was applied for chunking the noun phrases for each verse. A sample of bracketed structure and a tree is shown in Figure 5.2 and Figure 5.3. The parser receives a tagged sentence with a set of syntactic patterns and output a tree. The set of syntactic patterns was based on our analysis followed the pilot study for discovering patterns from the Qur’an explained in Section 5.1.1 and explained in [125]. The following figures give an example of parsing a verse number (1:4) into terms using the grammar listed above.

```
(S
        (NP1
                (IN ملك/N)
                (IN يوم/N)
                (DN ال/DET دين/N)))
```

Figure 5.2 A sample of bracketed structure produced by the shallow parser

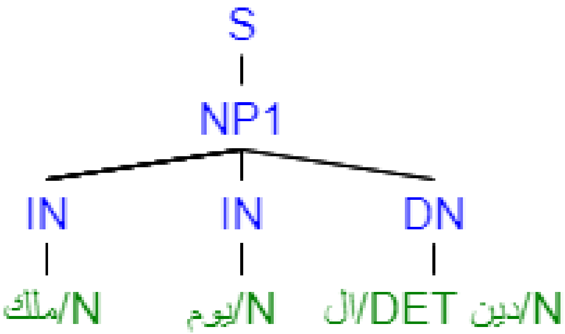


Figure 5.3 The parsed term depicted in a tree form.

A number of multi-word candidates which satisfied the provided patterns, but did not make correct stable lexical units were found in generating candidates based on only POS patterns. For example (“ المؤمنون الكافرون”. “the believers the disbelievers”), (“الناس السحر”, “the people the magic”), (“الجاهل أغنياء”, “the ignorant one self-sufficient”). (“زجاجة الزجاجة”, “a glass the glass”), (“خيفة موسى”, “a fear Musa”), and (“الهدى الشيطان”, “the guidance Shaitaan”) were found because patterns of “N-N”, “N DET.N” were include in the set of patterns. This occurs in verbless phrases, which composed of subject and predicate and they linked together using zero copula. It is also known as objective complement. Therefore, another feature for detecting stable lexical unit may improve the result.

5.1.2.1.2 Case-based Filter

This filter runs after the parser outputs the parsed tree. The parsed tree has no information about other features for the segments. We found the morphological case feature is useful for distinguish the subject predicate structure from phrases that makes a valid terms. Algorithm 5.1 shows more details and examples found in the text. In addition, agreement constraints in case are common in Arabic terms. Furthermore, [84] reported that case feature gives better results when they used with POS in Czech language. A similar results for Arabic reported in [85] for using case in their work. Another work who claims that morphology is essential syntactic modelling is in [83].

```
Input: $l$ ← a list of segments of white space separated,
$l$ = $\{s_1\, s_2 \dots s_n\}$
Output: cClass holds the concatenated sequences
function caseFilter($tmp, order1, case\_values$)
        cClass ←['false']* len(tmp)
        Conn ← hold database connection
        foreach $i$ ∈ range($tmp$) do
            units_candidates ← re.split('/| ', tmp[i].strip())
            foreach ib, bb in enumerate(case_values):
                flag=""
                for idx, o in enumerate(order1):
                    if(len(units_candidates)>1):
                         case1=bb.split(' ')
                          sql =\ "SELECT trim(case1)
                          from segments 13:WHERE sid = "
                          + units_candidates[o]
                          cursor.execute(sql)
                          ccc= getquery(cursor)[0]
                    if ccc[0]==case1[idx] and cClass[i]=='false':
                         flag=flag + case1[idx]+" "
              if(flag.strip()==bb):
                  cClass[i]='true'
        return cClass
```

Algorithm 5.1 The algorithm of filtering MWTs using case

In order to get the morphological for each segment, we use the ordinal position of the segment in the corpus instead of words in parsing at this time as it can be seen in the following in Figure 5.2 and Figure 5.3.

```
(S
      (NP1
            (IN 19/N)
            (IN 20/N)
```

(DN 21/DET
22/N)))

Figure 5.4 The bracketed structure with the ordinal position instead of words

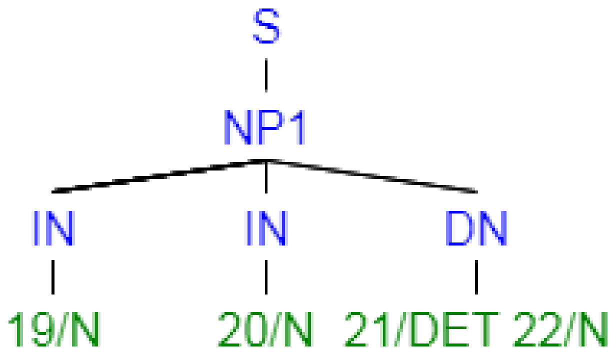


Figure 5.5 An example of a tagged verse with its output tree using the shallow parser.

Sequences
NOM-ACC
GEN-ACC
NOM-NOM
GEN-GEN
ACC-ACC

Table 5.3 Two sequenced case-based values for valid MWTs of noun-noun

After a verse being parsed, the case-based filter is run, which takes the morphological features for each segments as a vector and check the correspond feature for parsed segments. If they matched the predefined vector then the sequences tagged with true, otherwise false. Table 5.3 shows vectors of two elements the sequences of case-based applied in our filter. Note that the values of this vector was chosen based on our experiment on mining a number of MWTs that made of two terms shown in Table B.2 in Appendix B. We have collected the valid terms and get their case features and these values are related to the valid terms. This filter was applied for terms of two because they are of the majority of the MWTs and the values was also based on an experiment of MWTs of two terms.

5.1.2.1.3 Concatenation

Concatenation step is required because we used Qur'anic Arabic Corpus which is segmented into morphemes not words. Unlike words in other languages which they are delimited by space or punctuation marks, word in Arabic can be attached to another word. In Arabic clitics are attached to a stem or to another clitic without using orthographic marks such as apostrophe in English [126]. A word can be made of one or more morphemes and morphemes belong to a single word must be concatenated. In this step, we used clitic information from QAC to make a decision on whether should a sequences of extracted candidate concatenating their morphemes or not. We feed the parser with segment id with syntactic tags instead of text and then find information about the segment by its id, information included segment text and clitic information. Because same segment can be used in other places in different clitic form, we cannot rely on the text only. Algorithm 5.2 was developed for this step.

```
Input: $l$ ← a list of segments of white space separated,
$l$ = {$s_1$ $s_2$ … $s_n$}
Output: str holds the concatenated sequences
function Segmented2Concatenated ($l$)
        str ← “”
        foreach $s_i$ ∈ $l$ do
                if $s_i$ is “SUFFIX”
            str ← str.left(0,len(str)-1)
            str ← str + $s_i$ + “ ”
        elseif $s_i$ is “PREFIX”
```

```
            str ← str + $s_i$    //do not add any white space
        else
            str ← str + $s_i$ + “ ” //add it at the end
    return str
```

Algorithm 5.2 Concatenation of sequence words.

As we explained that terms in Arabic is concatenated and the concatenation process is necessary when computing the times of their occurrences in the same corpus or others. A list of separated segments is received by the algorithm, then generate the concatenation form of the tem based on clitic information for each segment.

5.1.2.1.4 Normalisation

The normalisation step converts text to simple Classical Arabic spelling system before computing measurements in the next phase. In addition, it removes diacritics and Hamza from letters. We found variation of spelling system used in previous Qur’anic Corpus. Previous results demonstrate that this normalisation step improves the accuracy of terms and concept extraction.

5.1.2.2 Domain-Specific Terms Extraction

This phase is measuring the important and relevance of a term to a given domain using a combination of linguistic and distributional methods.

This phase aims to compute the importance of a term. We use a hybrid scheme that considered different information from different sources as evidence to measure the importance of a term. This schema is able to measure single, multi-word and nested terms.

5.1.2.2.1 Weighting

In this step, a combination of statistical and domain-specific knowledge was created, based on formula (5.3), which was proposed by [26]. The choice of this method was based on the fact that it works well, even for a small text size and take into consideration MWTs. The statistical knowledge indicates the importance of a candidate in the text; simply, computing the relative frequency that a candidate t appeared in the corpus $p(t)$, as explained in formula number (5.1). The domain-specific knowledge, $w_d(t)$, is the number of times that t appeared as part of glossary list G, as described in equation (5.2). We chose the dataset of [51] because it is the only comprehensive topic list that is available in computer-processable form for the Qur'an.

$$P(t) = \frac{f(t)}{\max\limits_{1 \le i \le |TC|} f(t_i)} \qquad (5.1)$$

Where t is a candidate term, $f(t)$ is the number of times that candidate $t \in TC$ appeared in the corpus D, $\max\limits_{1 \le i \le |TC|} f(t_i)$ is the maximum number of term t that appeared in the corpus, and D, $P(t)$ is the statistical knowledge for a given t.

$$W_d(t) = 1 + \frac{\log\big(df(t)\big)}{\log\Big(\max\limits_{1 \le i \le |TC|} df(t_i)\Big)} \qquad (5.2)$$

In which $df(t)$ is the number of times that t appeared as part of a term in the glossary list G, $\max\limits_{1 \le i \le |TC|} df(t_i)$ is the maximum occurrences of t as part of another term from G, $W_d(t)$ is the domain-specific knowledge for a given t, and $|t|$ is the length of a term with regard to words number,

$$W(t) = \begin{cases} P(t) \times W_d(t), & if\ |t| = 1 \\ \sum_{i=1}^{|t|} W(t_i), & otherwise \end{cases} \tag{5.3}$$

where $\sum_{i=1}^{|t|} W(t_i)$ is the sum of the weight of each nested term $t_i\ of\ t$ if it was longer than one word. Computing nested terms is based on recursion technique shown in the Algorithm 5.3. And $W(t)$ is the total weight of a term.

Note that the function $f(t)$ in (**5.1**) is different from the function of $df(t)$ in (**5.2**) in terms of how a candidate information is computed. $f(t)$ calculates the appearance of the whole candidate in the corpus while $df(t)$ computes the appearance of the given candidate as part of the given corpus.

```
Input:  t ← a term, t = $w_1\, w_2 \ldots w_n$
function Weight_simple (t)
    if  t is in stopword
        return 0
    if t length is one word
        return P(t) × Wd (t)
    else
        //c ← P(t) × Wd(t)
        head ← is the first token of t
        tail ← is the rest of words in t
        return P(head) × Wd (head) + Weight(tail)
```

Algorithm 5.3 Nested terms weight calculation.

5.1.2.2.2 An Example of Weighting Calculation

Let us consider the term of ("تنزيل رب العالمين", "the Qur'an is the revelation of the Lord of the worlds.") from Table 5.4 as example to illustrate how the weights are calculated by the weighting schema by equation (5.3). The value of W("تنزيل رب العالمين") at the end of this illustration should be equal to 0.848644 according to Table 5.4. The weighting schema branches into two states depends on the length of the term. The first state applies when the length of term is one word. The second branch will be chosen if the length is more than one. Our selected term above is a multi-words term, therefore the second state will be chosen, which calculates the weight of all parts of the term in a recursive way. Each run the term is divided into two parts; head and tail until the length of tail is one.

In the first run, the term is divided to head and tail and the weight (W) is computed for the head of the term and the rest of the terms is added recursively as shown in the following example.

W(“تنزيل رب العالمين”) =
W(“تنزيل”)+ W(“رب العالمين”) , where
W(“رب العالمين”) =W(“رب”) + W(“العالمين”).

In order to compute the whole term the last word is calculated first: W(“العاملين”) is a single-word term, so the statistical knowledge and the domain-knowledge are computed to obtain the weight of this term.

The statistical part is calculated using equation (5.1), $P(t)$ which is $\frac{61}{2670}$ =0.022846442. The domain-knowledge is calculated by equation (5.2), $W_d(t) = 1 + \frac{\log(1)}{\log(87)} = 1$. The value of the W for the last part of the term now can be obtained using equation (5.3) which will implement the first state because the length of the term is one. $W(t) = 0.02284644 \times 1 =$ 0.02284644. The same processes apply to W(“رب العالمين”) which is equal to the $P(t) = \frac{1260}{2670}$ =0.471910112 and the $P(t) = \frac{15}{2670}$ = 0.005617978, now the value of W(“رب العالمين”) 0.843025866. Similarly the value of the whole term W(“تنزيل رب العالمين”) is computed using the W(“تنزيل”)+ W(“رب العالمين”) =0.005617978 + 0.843025866 = 0.848643844.

5.1.2.3 Concept Identification

In this phase, weighted terms are sorted by their weights, then the top-k terms are selected to represent the concepts of the domain. Finally, we validate the selected interval.

In order to find the most important terms we must find its relevance degree to the domain. The more domain-specific term is the more important it is. For example from Table 5.4, the weight of the term “تنزيل رب العالمين” in line 9 is higher than the term in line 10 which is “رب العالمين” although the occurrences of term in line 9 is less than term of line 10. Another example can be seen in the weight of the term “مالك يوم الدين” is 0.372031 which is higher than 0.28914 for the term “يوم الدين”. This indicates that the first term is more important than the second, even though the number of the occurrences of the second term is more and the first term is low frequency in the text but more important for ontology term extraction. Table 5.4 shows more examples in the results by giving the occurrences and W values for a list of terms. The term in the first row has only occurred twice but it is more specific to the domain than the term in row 9 and 10.

	Candidate	**Occurrences**	**Rel**	**W**
1	رسول رب العلمين	2	1	0.983648
2	لناس فى الكتب	1	0	0.946185
3	رب الناس	1	1	0.922347
4	رب موسى	2	1	0.909908
5	رب رحيم	1	1	0.905198
6	سبحن رب السموت	1	1	0.905198
7	رب السموت	13	1	0.889468
8	رب غفور	1	1	0.854262
9	تنزيل رب العلمين	1	1	0.848644
10	رب العلمين	42	1	0.843026
11	رب العرش العظيم	3	1	0.842996
12	الناس فى المهد	2	0	0.838125
13	رب هرون	1	1	0.831415
14	رب العرش الكريم	1	1	0.830636
15	رب العرش	6	1	0.829513

16	الءاخرة فى العذاب	1	0	0.826908
17	رب المشرق	5	1	0.824506
18	رب العزة	1	1	0.823176
19	رب الشعرى	1	1	0.820554
20	رب الفلق	1	1	0.820554
21	رب المغربين	1	1	0.820554
22	رب السمآء	0	1	0.820179

Table 5.4 Sample of ranked terms

The results included some terms with high weight according to functional words such as prepositions and conjunctions. Therefore, a stopwords list of functional words was used to prevent these words contributing to the increase of the weight of the terms.

5.1.3 Validation

In order to measure the accuracy of the model, we used three evaluation metrics that are commonly used in terms and concepts extraction. The first is comparing the result of the model against existing resources belong to the same domain. Second evaluation is based on two independent annotators who volunteered to do a manual extraction for a selected chapter. The third metric is computing Average precision (AvP), which is commonly applied in concept extraction researches such as in [69], [26].

All these metrics assess the accuracy of the extracted concepts in different ways. Validation against existing resources is the easiest in terms of implementation. No more human intervention needed and the overlap between the existing resource and the extracted is computed automatically. However, as it has mentioned in Chapter 3 the existing Qur’anic datasets are limited to a specific scope or cover few parts of the Qur’an. The second validation based on domain experts for partial validation. In this validation, a selected chapter from the extracted results was validated against the same chapter that covers from different existing resources. Third validation requires little human intervention for gives their feedback on the extracted list.

5.1.3.1 Available Ontologies

As it was mentioned in Chapter 3, most of available work for Qur'an has been done partly for some chapters. And there are only three resources available for the entire the Qur'an. Qurany, [48] which includes verses topics. Some terms in the Qurany have been written in the same meaning but using different words from the Qur'an. QurAna [41], primarily relied on pronouns mentioned in the Qur'an and linked them to a reference list composed of 1,028 concepts. These concepts only encompassed names or things that had been mentioned using pronouns, and did not cover those nouns that were not mentioned by their pronouns. Another dataset for Qur'anic concepts was established by Dukes & Atwell (2012) and Dukes (2013).

Related datasets	**Without normalisation**	**With normalisation**	
QAC	62.28%	77.34%	15.06
Qurany	20.65%	22.15%	1.5
QurAna	24.8%	57.01%	32.21

Table 5.5 Overlap with related resources

Table 5.5 shows the coverage percentages of the extracted concepts in related datasets. The overlap between the extracted list and the previous available datasets is computed for the three selected datasets. 62.28% of QAC concepts were found in the extracted list, and only 20.65% of Qurany and 24.8% of QurAna. The following analysis for the results revealed datasets based on spelling different from the spelling that was used for preparing the list. Therefore, A normalisation process was conducted for converting the list to the targeted spelling. We also removed vowels and Hamza characters from the datasets and the list. The results improved significantly with QurAna with 32.21 increase to the coverage percentage. We expected to the coverage with QAC higher than this 77.34% because most of its concepts are single word and using the same word not based on the meaning like Qurany or QurAna. However, abstract concepts of QAC are not mentioned in the Qur'an and cannot be extracted from its text such as "Artifact, Astronomical Body" and this is may cause the low in the overlap. Another problem is that we found some concrete concepts such as "سفينة نوح" "Noah's Ark" which is not found in our list because it is not found in the Qur'an in this format. Therefore, this validation is not appropriate in this case.

5.1.3.2 Partial Validation

We collected all available terms and concepts from previous work for the chapter 29. In addition, we asked two independent annotators to identify the concepts from the same chapter. Table 5.6 shows the comparison between these datasets in terms of how many of the concepts occurred in each verse. A1 is the annotation made by annotator1, while A2 is the data from annotator2. We also add our annotations that have discovered in the beginning in section 5.1.1 as A3.

Datasets	Terms	Unique terms
QAC	27	20
QurAna	324	48
Qurany	173	133
A1	497	348

A2	468	299
A3	354	248

Table 5.6 A comparison of different existing annotations from previous studies and manual annotations**.**

Table 5.6 illustrates the total number of terms found with previous datasets and through manual annotations. Human annotators identified more terms for a certain chapter of the Qur'an. This is because we did not ask the annotators to focus on a specific scope or pick certain patterns. QAC, QurAna, and Qurany are specialised for some specific proposes, which reveals why our extraction method did not achieve high precision in comparison with them.

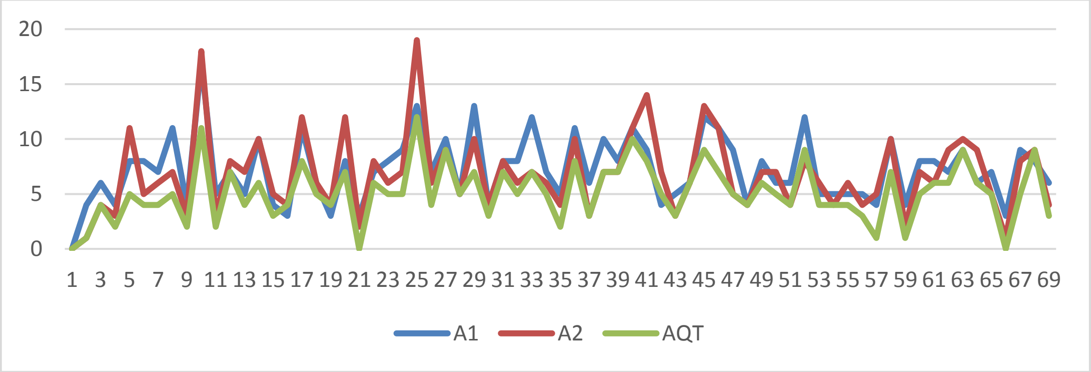


Figure 5.6 A comparison of hand-annotated terms.

Figure 5.6 shows the agreement between the two annotators who were asked to annotate the chapter 29 of the Qur'an. Although they carried out the task independently, the figure shows that they were very close together in terms of the numbers. However, this does not mean that their extracted terms for a certain verse are similar. We only focused on the number at this stage, to obtain a quick idea of how similar they were to each other.

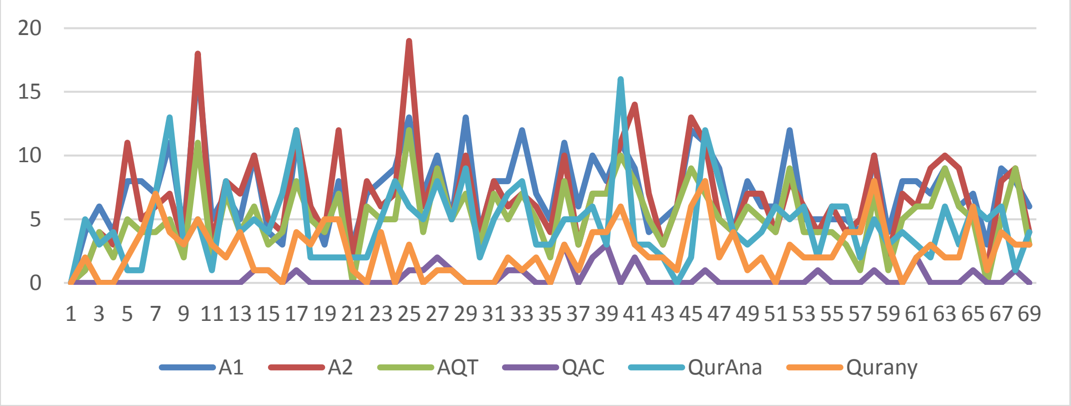


Figure 5.7 A comparison of hand-annotated and collected terms with those of previous related work.

This indicates that these datasets are not complete; therefore it is possible that our method may identify relevant terms that have not been covered in previous datasets. Therefore, a manual validation is needed for measuring the performance.

A manually validation to the extracted terms by a binary judgment that indicated which terms are relevant and which are non-relevant, after which we applied average precision (AvP). AvP is a very popular evaluation metric that is widely used to test the performance of term extraction methods and measuring the ontological knowledge extraction such as in [69] [26]. It is the sum of all precision to rank k over rank number (see equation (5.4)).

5.1.3.3 Average Precision (AvP)

AvP requires the extracted terms to be validated, therefore we put 1 for terms that are relevant to the Qur'an and 0 for non-relevant terms. Table 5.7 shows a sample of extracted terms with their validation values. For example, "{ld~iyna {loHamodu" are valid MWTs according to the POS pattern N N, but they do not make meaning, therefore they have given 0 for the column of Rel.

Arabic	Transliteration	Occurrences	Rel
الفلك المشحون	{lofuloki {loma$oHuwni	3	1

الملة الءاخرة	{lomil~api {lo'aAxirapi	1	1
العشى الصفنت	{loEa$iY~i {lS~a`fina`tu	1	0
الملا الاعلى	{lomala&lt;i {lo&gt;aEolaY`^	3	1
الوقت المعلوم	{lowaqoti {lomaEoluwmi	2	1
الدين الخالص	{ld~iynu {loxaAliSu	1	1
الملك اليوم	{lomuloku {loyawoma	2	0
الارض الفساد	{lo&gt;aroDi {lofasaAda	1	0
الملك اليوم	{lomuloku {loyawoma	2	0
الدين الحمد	{ld~iyna {loHamodu	1	0
العذاب المهين	{loEa*aAbi {lomuhiyni	2	1

Table 5.7 The validated terms with their validation binary value in the most right column.

$$AvP = \frac{\sum_{k=1}^{n}(p(k) \times rel(k))}{n_c} \qquad \textbf{(5.4)}$$

Where $p(k)$ is the precision at cut-off k in the terms list, n means the size of the extracted terms list, n_c is the total number of relevant terms that were retrieved by the method and $rel(k)$ is a binary function that indicated whether or not the retrieved term was relevant. The output of $rel(k)$ is 1 if a $term_k$, which means the term at k, is relevant to the Qur'an domain and 0 otherwise.

$$R@k = \frac{the\ number\ of\ relevant\ retrieved\ at\ rank\ k}{all\ relevant\ retrieved} \qquad \textbf{(5.5)}$$

$$P@k = \frac{the\ number\ of\ relevant\ retrieved\ at\ rank\ k}{number\ of\ relevant\ and\ non-relevant\ retrieved\ at\ rank\ k} \qquad \textbf{(5.6)}$$

Where $R@k$ is the recall at rank k and $P@k$ is the precision at rank k. In order to check the performance of a method AvP-rank curve or precision-recall curve is plotted and assessed by looking at the relationship between them. Good method that extract a list which ranks relevant terms near to the top of the ranking list [26].

k	*recall*	*precision*	*AvP*
1	0.001789	1	1.000000
50	0.06619	0.74	0.778600
100	0.144902	0.81	0.784534
150	0.211091	0.786667	0.790667
200	0.289803	0.81	0.793123
250	0.357782	0.8	0.796795
300	0.402504	0.75	0.792789
350	0.457961	0.731429	0.785697
400	0.516995	0.7225	0.778722
450	0.567084	0.704444	0.771061
500	0.601073	0.672	0.762724
550	0.65653	0.667273	0.754340
600	0.695886	0.648333	0.746516
650	0.726297	0.624615	0.737926
700	0.749553	0.599428	0.728578
750	0.801431	0.598131	0.719892
800	0.844365	0.590738	0.711766
850	0.876565	0.57715	0.704337
900	0.892665	0.555061	0.696761
950	0.942755	0.555321	0.689081
1,000	1	0.55956	0.682584

Table 5.8 The average precision for the first 1,000 terms in our list.

This table shows the AvP of the top 1000 extracted terms, and we can clearly observe that those ranked nearest to the top had high precision, which then decreased in accordance with the increase in the size. Recall increased as the number of candidates rose, while precision decreased. We obtained an overall precision of 0.81 for the first 200 terms.

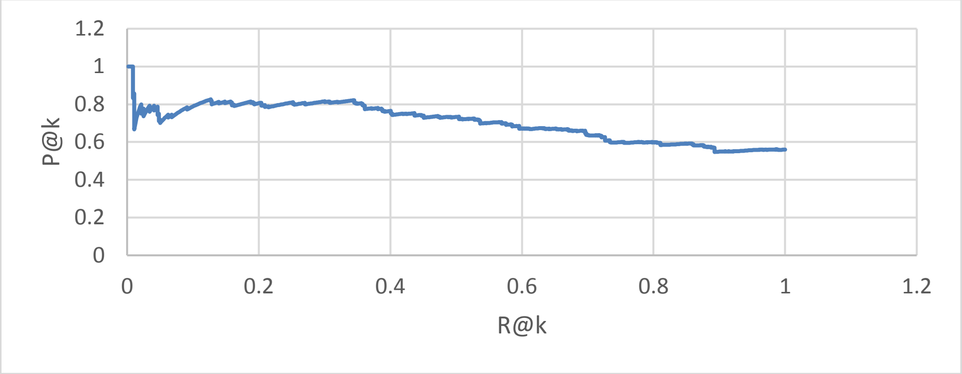


Figure 5.8 Recall-precision graph for the first 1,000 extracted terms.

Figure 5.8 shows the precision for every k in the list. The precision associated with the candidates at the very top is higher than the precision at the bottom

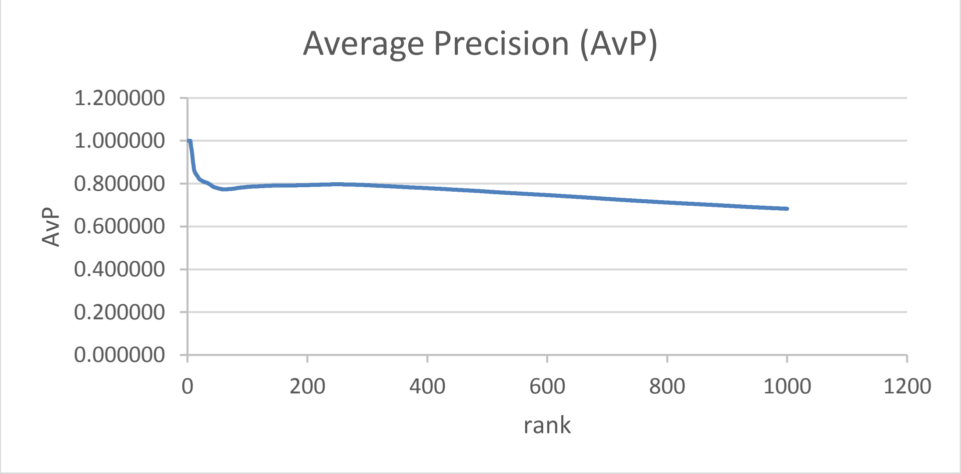


Figure 5.9 The AvP-rank curve for the identified concepts.

Figure 5.9 shows the relationship between precision and rank number. The graph represents a natural way of looking at the extracted list performance at every position in the ranking list. This graph clearly shows that our methods achieved approximately 0.65 as overall precision and 0.8 precision for the first 200 candidates.

5.2 Further Experiments and Evaluation

In order to evaluate the performance of the improvements, an experiment was carried out to test it by comparing the extracted terms based on three different way of candidate generating.

Method	**Accuracy**			
Traditional patterns	**K**	**R@K**	**P@K**	**AvP**
	50	0.068354	0.55102	0.655342
	100	0.124051	0.494949	0.592061
	150	0.197468	0.52349	0.567891
	200	0.278481	0.552764	0.558542
Predefined patterns	**K**	**R@K**	**P@K**	**AvP**
	50	0.170854	0.68	0.669238
	100	0.346734	0.69	0.667427
	150	0.527638	0.7	0.682052
	200	0.758794	0.755	0.694534

	K	R@K	P@K	AvP
With case-based filter	50	0.052689	0.96	0.927611
	100	0.107574	0.98	0.950042
	150	0.15258	0.926667	0.945154
	200	0.198683	0.905	0.94064

Table 5.9 Comparison between three methods for Concept extraction

K	AvP			AvPDiff
	V1	V2	V3	
50	0.655342	0.669238	0.927611	+ 0.272269
100	0.592061	0.667427	0.950042	+ 0.357981
150	0.567891	0.682052	0.945154	+ 0.377263
200	0.558542	0.694534	0.94064	+ 0.382098

Table 5.10 A comparison between first attempt and the second attempt in Concept identification.

Table 5.10 shows the summary of a comparison between the three experiments in identifying the set of concepts. We divided the retrieved terms into 4 groups of k; top-50, top-100, top-150 and top-200. Then calculate the precision according to Equation (5.6), recall according to Equation (5.5) and AvP based on Equation (5.4). Table 5.10 shows that AvP at the third experiment has increased at every point of k, a 0.3 improvement in AvP compared to prior work on the same task. In addition, Figure 5.10 compares the results from the three experiments and depicts the improvement achieved at the last experiment. The last experiment, which uses the predefined patterns with inflectional information outperforms the other two which conclude that combining inflectional information within patterns get better results in identifying concepts. On the other hand, the manual validation is needed due to the lack of available resources that can be compared with.

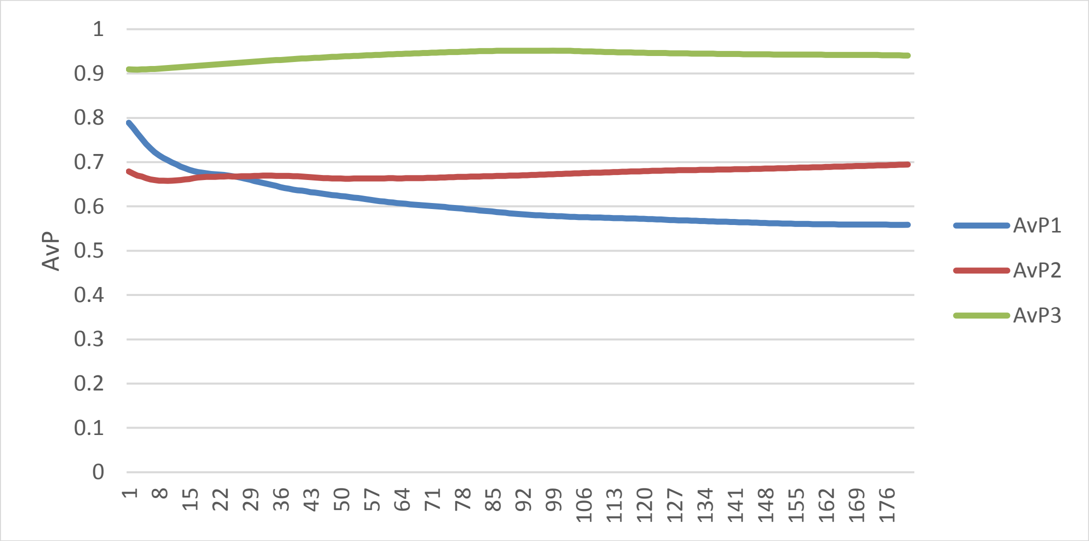


Figure 5.10 The average precision for the three experiments

Figure 5.10 compares three methods of identifying concepts by plotting their curves based on AvP distributed over number of points. As can be seen in Figure 5.10, V1 showed decreasing retrieval performance compared to V2 and V3. The set of terms extracted based on the predefined patterns (V2), obtained an improvements in retrieval performance compared to V1 which was based on traditional patterns. V3 is the same as V2 but with applying a filter based on morphological feature “case” at the candidate generation phase.

5.3 A supervised method for Qur’anic term extraction based on AQT dataset

In this section, we conducted a number of experiments using a software called WEKA [127], which combines a collection of machine learning algorithms for data mining. It also provides a number of filters that deal with datasets pre-processing such as converting text into word vectors.

We use the validated extracted terms explained above in this chapter. The goal is to enhance the domain-specific terms extraction and compare it to unsupervised method. The experiment was run using a training set of 222 and test set were selected randomly and the number of positive and negative instances were (approximately) equal.

5.3.1 Features Selection

We included a number of features with each instance in the ARFF file, which is the file used to train the model of extracting compound terms from the Arabic text of the Qur'an.

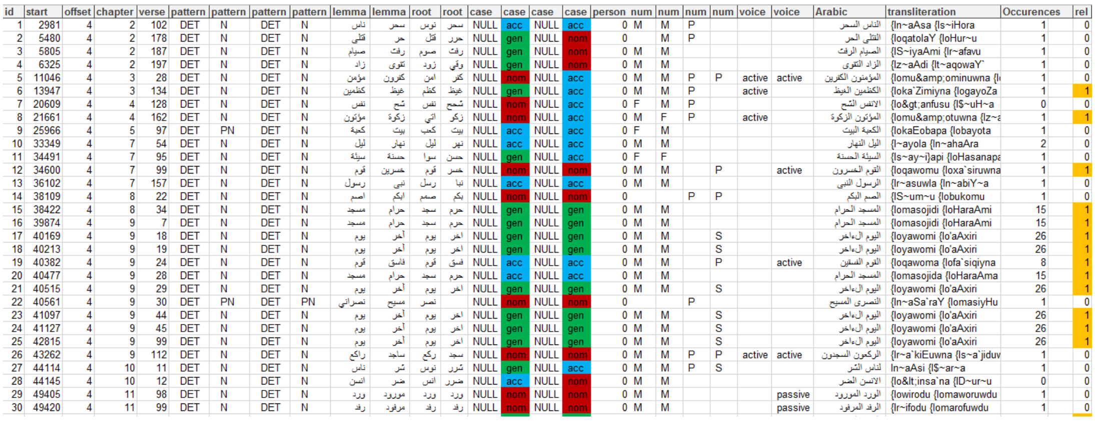
- Lexical Features (**L**EX)
  posi : This includes the part-of-speech tag for the segment in position i in the current term. We add this feature to the term because it is commonly used for the stage of phrases selection in most of term extraction methods. Examples of multi-words terms candidate selection based on POS tags can be seen in [37], [70], [76].

- Morphological Features (**M**ORPH)
  Inflectional morphological information of the noun
  casei : A number of multi-word terms were incorrectly classified as terms while they are objective complement or verbless copula construct, which links between the subject and the predicate of a sentence. This kind of construction can be distinguished from a complete phrase using case information. For instance, the subject in the Arabic language in most cases is nominative and the predicate is also nominative in verbless sentences [128]. For example Table 5.11 the row no 2,3 and 3 are not terms and have the same sequence of case information “genitive-nominative”.

- Derivational morphological information of the noun
  genderi : This feature is an important for numbers where first noun gender is different from the followed noun.

- Num : This features found in nominal segments such as noun and adjectives.
- Voice : is a feature of two values : active or passive

- Occurrences : the count of segments. It shows the frequency of the segments in relation of the whole corpus.
- W : is the weighting score calculated for each term based on formula (5.3) in this chapter.



id	start	offset	chapter	verse	pattern	pattern	pattern	pattern	lemma	lemma	root	root	case	case	case	case	person	num	num	num	num	voice	voice	Arabic	transliteration	Occurences	rel
1	2981	4	2	102	DET	N	DET	N	ناس	سحر	نوس	سحر	NULL	acc	NULL	acc	0	M	M	P				الناس السحر	{ln~aAsa {ls~iHora	1	0
2	5480	4	2	178	DET	N	DET	N	قتلى	حر	قتل	حرر	NULL	gen	NULL	nom	0		M	P				القتلى الحر	{loqatolaY {loHur~u	1	0
3	5805	4	2	187	DET	N	DET	N	صيام	رفث	صوم	رفث	NULL	gen	NULL	nom	0	M	M					الصيام الرفث	{lS~iyaAmi {lr~afavu	1	0
4	6325	4	2	197	DET	N	DET	N	زاد	تقوى	زود	وقي	NULL	gen	NULL	nom	0	M	M					الزاد التقوى	{lz~aAdi {lt~aqowaY`	1	0
5	11046	4	3	28	DET	N	DET	N	مؤمن	كفرون	امن	كفر	NULL	nom	NULL	acc	0	M	M	P	P	active	active	المؤمنون الكفرين	{lomu&amp;ominuwna {l	1	0
6	13947	4	3	134	DET	N	DET	N	كظمين	غيظ	كظم	غيظ	NULL	gen	NULL	acc	0	M	M	P		active		الكظمين الغيظ	{loka`Zimiyna {logayoZa	1	1
7	20609	4	4	128	DET	N	DET	N	نفس	شح	نفس	شحح	NULL	nom	NULL	acc	0	F	M	P				الانفس الشح	{lo>anfusu {l$~uH~a	0	0
8	21661	4	4	162	DET	N	DET	N	مؤتون	زكوة	اتي	زكو	NULL	nom	NULL	acc	0	M	F	P		active		المؤتون الزكوة	{lomu&amp;otuwna {lz~a	1	1
9	25966	4	5	97	DET	PN	DET	N	كعبة	بيت	كعب	بيت	NULL	acc	NULL	acc	0	F	M					الكعبة البيت	{lokaEobapa {lobayota	1	0
10	33349	4	7	54	DET	N	DET	N	ليل	نهار	ليل	نهر	NULL	acc	NULL	acc	0	M	M					اليل النهار	{l~ayola {ln~ahaAra	2	0
11	34491	4	7	95	DET	N	DET	N	سيئة	حسنة	سوا	حسن	NULL	gen	NULL	acc	0	F	F					السيئة الحسنة	{ls~ay~i}api {loHasanapa	1	0
12	34600	4	7	99	DET	N	DET	N	قوم	خسرين	قوم	خسر	NULL	nom	NULL	nom	0	M	M		P		active	القوم الخسرون	{loqawomu {loxa`siruwna	1	1
13	36102	4	7	157	DET	N	DET	N	رسول	نبى	رسل	نبا	NULL	acc	NULL	acc	0	M	M					الرسول النبى	{lr~asuwla {ln~abiY~a	1	0
14	38109	4	8	22	DET	N	DET	N	اصم	ابكم	صمم	بكم	NULL	nom	NULL	nom	0			P	P			الصم البكم	{lS~um~u {lobukomu	1	0
15	38422	4	8	34	DET	N	DET	N	مسجد	حرام	سجد	حرم	NULL	gen	NULL	gen	0	M	M					المسجد الحرام	{lomasojidi {loHaraAmi	15	1
16	39874	4	9	7	DET	N	DET	N	مسجد	حرام	سجد	حرم	NULL	gen	NULL	gen	0	M	M					المسجد الحرام	{lomasojidi {loHaraAmi	15	1
17	40169	4	9	18	DET	N	DET	N	يوم	آخر	يوم	اخر	NULL	gen	NULL	gen	0	M	M		S			اليوم الءاخر	{loyawomi {lo'aAxiri	26	1
18	40213	4	9	19	DET	N	DET	N	يوم	آخر	يوم	اخر	NULL	gen	NULL	gen	0	M	M		S			اليوم الءاخر	{loyawomi {lo'aAxiri	26	1
19	40382	4	9	24	DET	N	DET	N	قوم	فاسق	قوم	فسق	NULL	acc	NULL	acc	0	M	M		P		active	القوم الفسقين	{loqawoma {lofa`siqiyna	8	1
20	40477	4	9	28	DET	N	DET	N	مسجد	حرام	سجد	حرم	NULL	acc	NULL	acc	0	M	M					المسجد الحرام	{lomasojida {loHaraAma	15	1
21	40515	4	9	29	DET	N	DET	N	يوم	آخر	يوم	اخر	NULL	gen	NULL	gen	0	M	M		S			اليوم الءاخر	{loyawomi {lo'aAxiri	26	1
22	40561	4	9	30	DET	PN	DET	PN	نصراني	مسيح	نصر		NULL	nom	NULL	nom	0			P				النصرى المسيح	{ln~aSa`raY {lomasiyHu	1	0
23	41097	4	9	44	DET	N	DET	N	يوم	آخر	يوم	اخر	NULL	gen	NULL	gen	0	M	M		S			اليوم الءاخر	{loyawomi {lo'aAxiri	26	1
24	41127	4	9	45	DET	N	DET	N	يوم	آخر	يوم	اخر	NULL	gen	NULL	gen	0	M	M		S			اليوم الءاخر	{loyawomi {lo'aAxiri	26	1
25	42815	4	9	99	DET	N	DET	N	يوم	آخر	يوم	اخر	NULL	gen	NULL	gen	0	M	M		S			اليوم الءاخر	{loyawomi {lo'aAxiri	26	1
26	43262	4	9	112	DET	N	DET	N	راكع	ساجد	ركع	سجد	NULL	nom	NULL	nom	0	M	M	P	P	active	active	الركعون السجدون	{lr~a`kiEuwna {ls~a`jiduw	1	0
27	44114	4	10	11	DET	N	DET	N	ناس	شر	نوس	شرر	NULL	gen	NULL	acc	0	M	M	P	S			لناس الشر	ln~aAsi {l$~ar~a	1	0
28	44145	4	10	12	DET	N	DET	N	انسن	ضر	انس	ضرر	NULL	acc	NULL	nom	0	M	M					الانسن الضر	{lo<insa`na {lD~ur~u	0	0
29	49405	4	11	98	DET	N	DET	N	ورد	مورود	ورد	ورد	NULL	nom	NULL	nom	0	M	M				passive	الورد المورود	{lowirodu {lomaworuwdu	1	0
30	49420	4	11	99	DET	N	DET	N	رفد	مرفود	رفد	رفد	NULL	nom	NULL	nom	0	M	M				passive	الرفد المرفود	{lr~ifodu {lomarofuwdu	1	0

Table 5.11 an example of hybrid information annotated for each term and their references.

Table 5.11 Shows our dataset of multi-word terms along with the following features:

Table 5.11 shows several information for bigram terms used in machine learning experiment to predict the whether a given sequence is a valid term or not. First column, start, is the position in the corpus where this term start. Offset indicates the number of segments this term composed of. Chapter refers to the number that this term was mentioned in. Verse is the verse number in which this term was mentioned. Lemmas and roots were included for all segments of the term, thus there are a number of lemmas and roots equal to the offset. Arabic is the term in Arabic language, and transliteration includes the term in Buckwalter. Occurrences is the number of times that this term occurs in the corpus. Rel is a binary validation where each term manually verified whether or not the retrieved term was relevant, 1 for correct and 0 for incorrect. W is the calculated weight for each term according to equation (5.3) In this chapter.

5.3.2 Classification

We divided the AQT dataset into two files; training and test. The classifier J48 was selected for learning rules of extraction. The following decision tree in Figure 5.11 shows an example of Weka's visualisation of the built model using the aforementioned features. Occurrences of the terms were included which is the model started with.

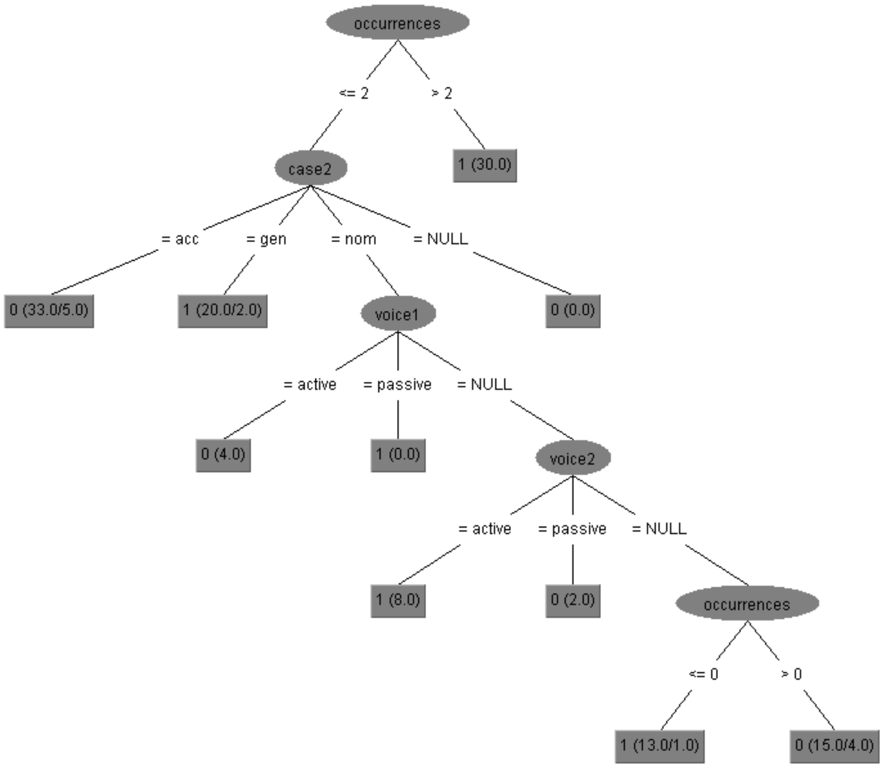


Figure 5.11 Decision tree visualisation output for the created model using J48 algorithm.

As it can be seen, tree start branching on the feature occurrences and the second branch is the case information for the second part of the term. Therefore, this an indication that these features are very important to be considered in terms selection.

5.3.3 Extraction Results

J48 correctly classified with an accuracy of 84%. Table 5.12 shows the results for two extraction experiments.

The learning curve of the generated model is plotted in Figure 5.12. the y represents the error rate and x represents the percentage of data that were taken out of the training data. The curve shows the worst case of the model when most of data were taken out and the best performance is when only 10% of data taken out.

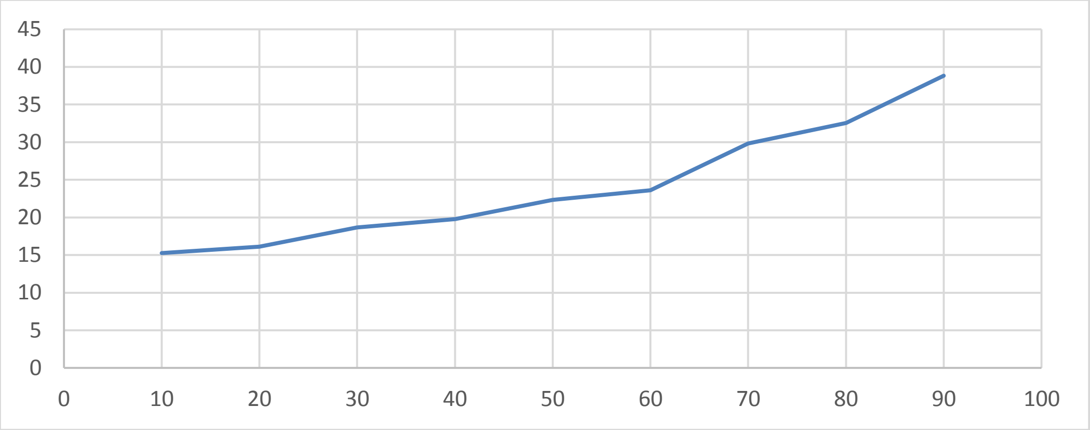
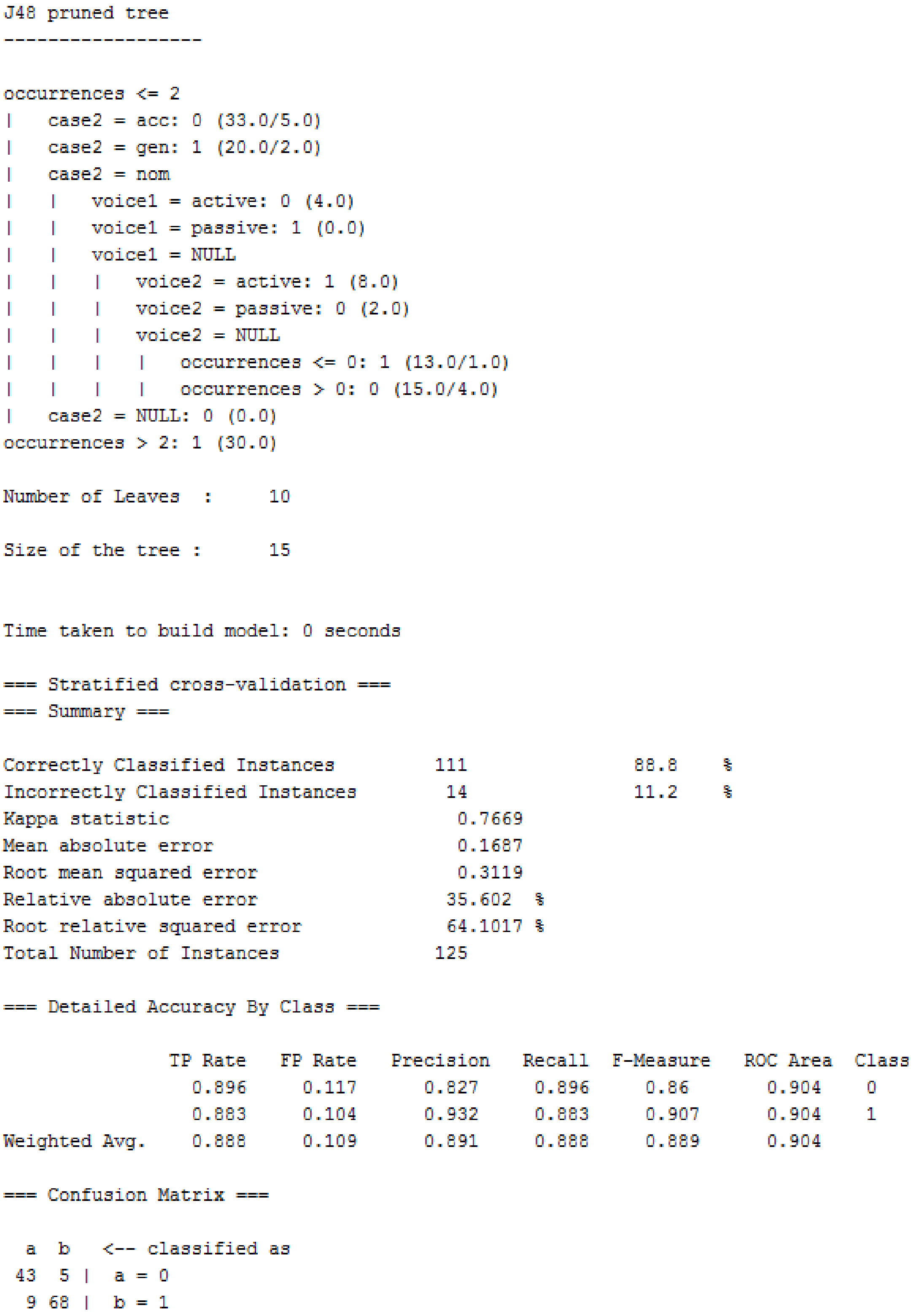


Figure 5.12 The learning curve of the generated model.



```
J48 pruned tree
------------------

occurrences <= 2
|   case2 = acc: 0 (33.0/5.0)
|   case2 = gen: 1 (20.0/2.0)
|   case2 = nom
|   |   voice1 = active: 0 (4.0)
|   |   voice1 = passive: 1 (0.0)
|   |   voice1 = NULL
|   |   |   voice2 = active: 1 (8.0)
|   |   |   voice2 = passive: 0 (2.0)
|   |   |   voice2 = NULL
|   |   |   |   occurrences <= 0: 1 (13.0/1.0)
|   |   |   |   occurrences > 0: 0 (15.0/4.0)
|   case2 = NULL: 0 (0.0)
occurrences > 2: 1 (30.0)

Number of Leaves  :     10

Size of the tree :      15


Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         111               88.8    %
Incorrectly Classified Instances        14               11.2    %
Kappa statistic                          0.7669
Mean absolute error                      0.1687
Root mean squared error                  0.3119
Relative absolute error                 35.602  %
Root relative squared error             64.1017 %
Total Number of Instances              125

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.896     0.117      0.827     0.896     0.86       0.904    0
                 0.883     0.104      0.932     0.883     0.907      0.904    1
Weighted Avg.    0.888     0.109      0.891     0.888     0.889      0.904

=== Confusion Matrix ===

  a  b   <-- classified as
 43  5 |  a = 0
  9 68 |  b = 1
```

Figure 5.13 2-gram terms classification.

Feature	**Algorithm**	**Precision**	**Recall**	**F1-Measure**
Without	J48	0.856	0.840	0.843
With W	J48	0.856	0.840	**0.843**
	LibSVM	**0.872**	0.840	0.826
N DET.N	J48	**0.982**	0.991	0.986

Table 5.12 Accuracy measures for term extraction based on different feature sets.

In order to compare the results of this experiment with the previous results in Table 5.9, the results of the set V3 which generated using cas-based filter has obtained 0.905 precision for the top 200 retrieved candidates. It can be clearly seen this experiment has achieved lower precision for both algorithms J48 and LibSVM. However, the results are better than V1 which obtained by traditional patterns. The precision of the set V1 was 0.552764 which we only used the traditional POS pattern in the selection stage. The set V2, which predefined patterns used for generation has achieved 0.755 in terms of precision. In set V3, we use case-base filter in order to assess the validity of a candidate and similar feature was used for training the supervised model, while others has not been generated according to this feature. This may reveal why other sets like V1 and V2 has achieved lower precision.

5.4 Summary

This chapter proposed a pipelined architecture for identifying concepts from the Arabic text of the Qur'an. This work is distinguished from previous research in Arabic concept identification as it uses lexical and inflectional information for terms candidate extraction. In addition, it uses different schema from previous work for ranking these terms, which computes a combination of domain-specific and statistical for weighting and extracting the domain-specific terms. The extracted concepts of this architecture were evaluated using three evaluation metrics. The first metric was based on datasets from previous studies of the Qur'an which were used as a gold-standard. This evaluation has obtained low performance due to the variation in scope and coverage they based on. The higher coverage got was in comparison with QAC is 62.28%. Although we tried to normalise the selected dataset into the same spelling format with the extracted list, the results have a little increase with 77.34%. The reason for that is the manual extracted concepts in these datasets does not represent the actual words occur in the text. Partial evaluation was conducted for a certain chapter that was covered by selected datasets and volunteer annotators. The results show an agreement with hand-annotated terms and more close to them than the collected from selected datasets. Finally, the extracted list was validated and AvP was calculated. Three experiments using AvP were conducted for improving the precision of the extracted results. The initial results was based on traditional POS patterns for term extraction. This achieved precision of up to 0.55 for the top 200 terms. The second attempt of our experiment was based on a predefined patterns, it improved the precision to 0.755 for the same size of terms. The third one was based on the predefined with inflectional. The performance was increased to 0.905. As a result, this chapter provided a resource called AQT to be used in evaluation and as a training for supervised concept identification systems.

Finally, it presented possible applications to the work presented in this thesis. An experiment of applying machine learning technique has been conducted for learning Qur'anic domain-specific terms. We exploit the resource AQT which was manually validated to build a model using machine learning algorithm. The result is compared to the previous results.

Chapter 6

Hierarchical Structure

Hierarchical relations or taxonomies are the backbone and the main components of ontologies [6], [87]–[89]. Taxonomies construction is a task of extracting pairs of terms linked by hypernym-hyponym ("is-a") relations. One of the major challenge in ontology development is automatic discovery of taxonomies. Automatic discovery of "is-a" relations has been studied under different names such as concept hierarchies and taxonomy acquisition, but we will use the hierarchical structure term in this thesis for denoting the automatic discovery of "is-a" relations from the Arabic text of the Qur'an. This chapter presents our methodology of automatic hierarchical relations construction among identified concepts from the Chapter 5. Our algorithm exploits the information of internal structure of multi-word terms, the head-modifier principle for extracting the hierarchical candidates of multi-word terms. In this method, the head of a multi-word term is assuming the hypernym role [93]. Our algorithm also takes into consideration other types of terms that their heads do not make a good hypernym such as multi-word terms headed with non-nominals. Therefore, a specific linguistic construct, called copula that links the predicate to the subject in an equational sentences [104], was employed for this type of terms.

To the best of our knowledge, this is the first work proposes a method for extracting inexplicit "is-a" relations for the Arabic language based on pronominal information. And presenting a hybrid methods combine Head-Modifier and a number of Copulas patterns for extracting is-a relationships from Arabic text.

6.1 Algorithm Overview

This section describes our algorithm for extracting hypernym-hyponym relations among domain-specific terms were extracted in Chapter 5. The presented algorithm composed of two main components for elicit hierarchal relations. First component is based on a methodology similar of [92], [129] for single-word and multi-word terms, taxonomy relations among terms headed with noun are extracted. The second component attached to the algorithm is pattern-based which relies on a specific pattern occurs in the text called copula construct. The copula construct occurs in various way in Arabic language (i.e. pronoun, zero copula and the verbal copula). The following Algorithm 6.1 gives an overview of how it works then we describes the details for each components.

Let T be a set of terms that we want to find their hierarchal relations. Each term t is constructed of a sequence of lexical segments w_i:

$$T = \{t_1, t_2, \cdots, t_n\}\ m = |T|$$

$$t = \{w_1, w_2, \cdots, w_n\}\ n = |t|$$

Whereas T is the set of domain-specific terms extracted from previous chapter. We based on the delivered list namely AQT from Chapter 5, which extracted based on the shallow parser and morphological features used to extract valid lexical units of words.

$$\forall\ t_i\ \exists\ \ t_j : lem(\ t_j) = lem(w_1) \in t_i$$

$$: hyponym\big(lem(t_i), t_j\big)$$

$$i \le 1\ \ \le (|T| - 1)$$

$$j \le\ \ i + 1\ \ \le |T|$$

head(t_i) or w_1 is the head of given term t lem(t_i) is a function of a lemma of a given word t

```
For $i \leftarrow 1\ to\ n - 1$
    For $j \leftarrow i + 1\ to\ n$
        If $|t_i| > 1$, head($t_i$) is an indefinite noun then
            If $lem(\text{head}(t_i)) = lem(t_j)$
                Add : $\boldsymbol{hyponym}(lem(t_i), t_j)$
            End if
        Else if $t_i$ copula $t_j$ then
            Add : $\boldsymbol{hyponym}(parent(t_i), t_j)$
        End if
    End for
End for
```

Algorithm 6.1 Hyponyms extraction based head-modifier and copula

6.1.1.1 Head-Modifier

One of the most common methods to elicit concepts with hyponym-hypernym relation is to use internal structure information of multi-word concepts by comparing their substring representations with each other. For example, “oat meal” is a hyponym of “meal”. Since most of the Qur’anic terms are made of multi-word terms with a percentage of more than 60%, Head-Modifier methods may find many hyponym-hypernym relations among them. However, Arabic compounds terms are left headed unlike English. Using available tool for English my requires significant modifications to work for Arabic. Another different is English prefix may considered as terms and compared to other terms such as “pin” and “candlepin” in [92], [129]. For Arabic prefixes do not make a term. But for suffixes it can refer to something such as pronouns. For this we applied idea of multiword terms of [129].

We included this way of relation extraction as a component in our work. This component compares compound terms with single-word terms. It takes into consideration the nested compound terms that headed with nominals. In this component 2919 terms were linked to their head as hyponyms based on compared to each other. Because Arabic is highly inflected language, some concepts take different spelling as their morphological features is changing. For example, like “المسلمين”, “المسلمون” which belong to the same concepts but they were mentioned in various spelling. Hyponyms relations between the lemma of the head of a term and its modifier are added each time that the algorithm find two terms that one of them has occurred as a head with the other one. Lemma was used to normalise the variation of way that the same term is used in the text. The first argument of the hyponym function represent the hypernym and the second represent the hyponym. Figure 6.1 illustrates an example of partial ontology produced by this method for the hierarchy built from the noun phrases with the head “the day”. In this example we show the concept of “the day”. As it can be seen in the following figure, a number of terms that share the same lemma of the term day were assigned to it as hyponyms. We also found a number of examples can be seen in Table 6.1 and Table 6.2.

Due to not all single terms have a significant occurrences in the Qur’an we started by grouping concepts based on multi-words term structure by recursively dividing multi-words term into two parts; head and modifier. Then, take the lemma of the head as the hypernym of the given term. After that, we divided modifiers into head and modifier until the length of the term become one word. The reason for recursively dividing terms, is that our constructed terms have many nested terms, which other terms are nested in. Algorithm 6.2 shows how we implemented this linking.

```
Input：  t ← a term, t = $w_1 w_2 ... w_n$
function triples (t)
    if  t is in stopword
        return null
    if t length is one word
        return null
        if $w_1$ is not nominal
            return null

    head ← is the first token of t
    return (t, is-a, head)
```

Algorithm 6.2 Recursively linking multi-word terms to their head

As a result of applying this algorithm we found many terms linked in many different ways such as nested term linked with their head. Table 6.1 shows an example of the individuals of a certain term. These individuals linked directly with the term “the day of Judjment” and this term and other terms headed with same lemma linked together under the head “Day” such as “yawmi l-dīni” and “The last day” as it can be seen in the Figure 6.1.

id	start	chapter	verse	Arabic	transliteration	ref
69	11832	3	3	يوم القيمة	yawomi {loqiya`mapi	80
98	14840	3	3	يوم القيمة	yawoma {loqiya`mapi	80
122	19169	3	4	يوم القيمة	yawomi {loqiya`mapi	80
151	22963	3	5	يوم القيمة	yawomi {loqiya`mapi	80
159	23721	3	5	يوم القيمة	yawomi {loqiya`mapi	80
166	24842	3	5	يوم القيمة	yawomi {loqiya`mapi	80

Table 6.1 Sample of individual occurences for the concept of “the day of Judjment”

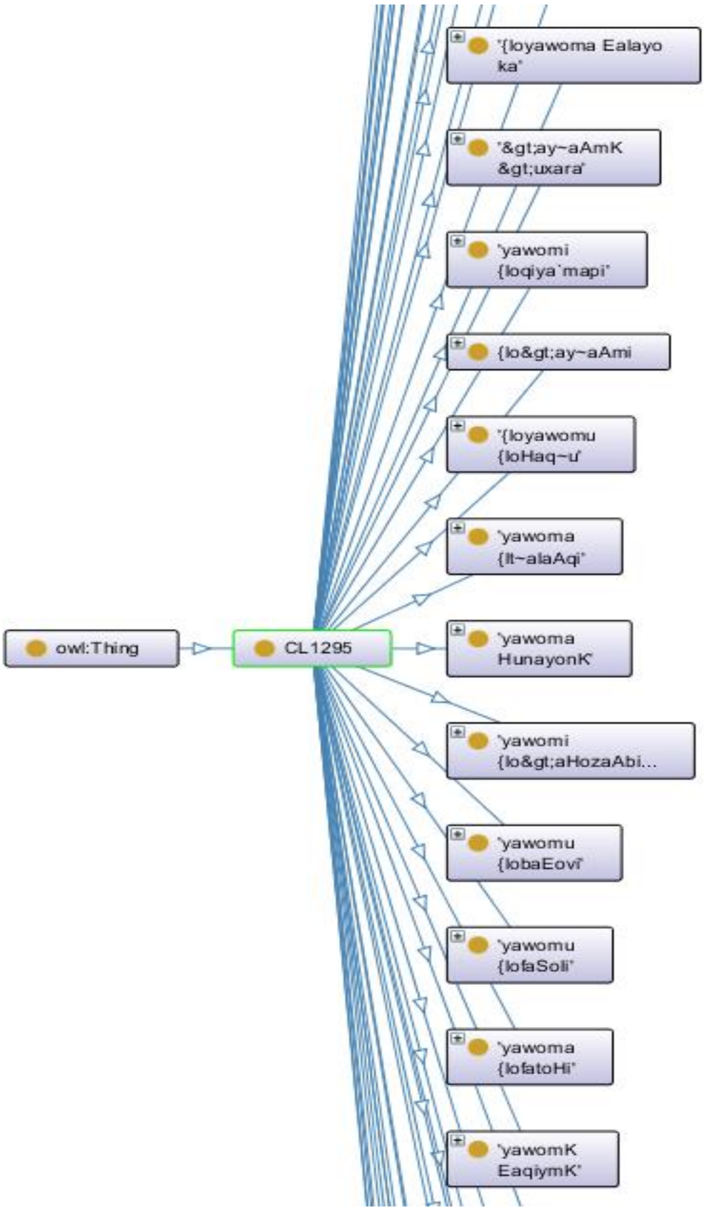


Figure 6.1 A partial ontology consists of the concept day and hierarchy.

Table 6.2 shows the terms number, label and their reference to the head they should linked to and the id for similar concepts from external ontologies. For example most of the term linked to of 1294, which refers to the "Day". Another point to note is these terms are linked to the term "Day" regardless their inflection variation.

			link to	link to
id	Arabic	reference	QAC	Qurany
1812	يـئوس كفور	1296	#N/A	#N/A
2294	يونس	1295	198	#N/A
3	ايام الله	1294	#N/A	#N/A

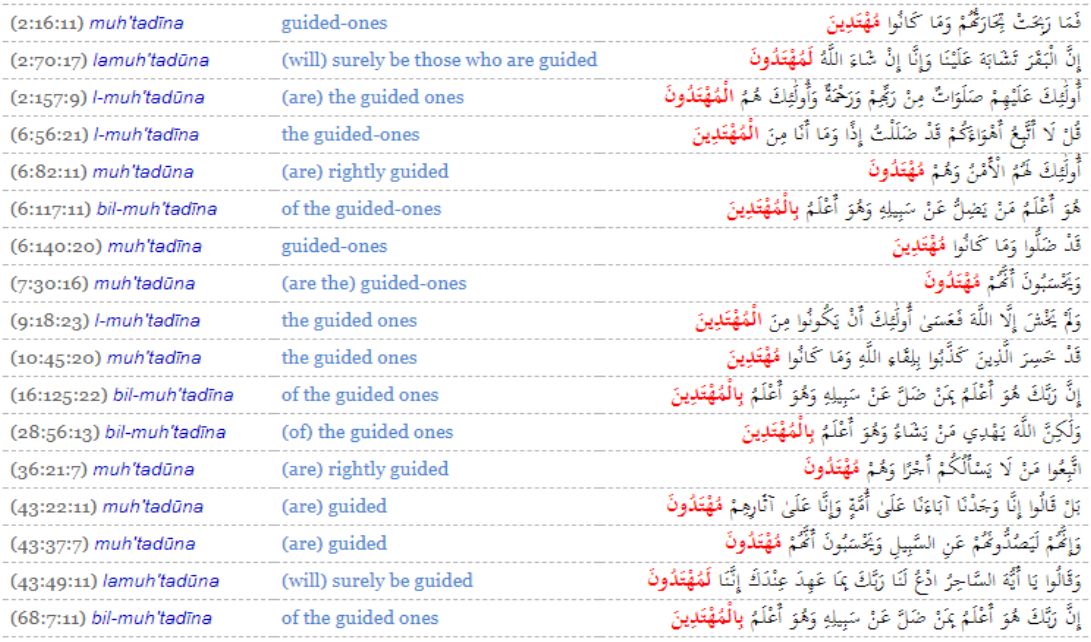
17	ايم الله	1294	#N/A	#N/A
23	يوم عظيم	1294	#N/A	#N/A
37	يوم كبير	1294	#N/A	#N/A
40	ايام اخر	1294	#N/A	#N/A
48	اليوم الءاخر	1294	#N/A	#N/A
62	يوم اليم	1294	#N/A	#N/A
76	يوم الحج الاكبر	1294	#N/A	#N/A
80	يوم القيمة	1294	#N/A	#N/A
86	اليوم علي ك	1294	#N/A	#N/A
90	يوم الدين	1294	#N/A	#N/A
91	يوم الحج	1294	#N/A	#N/A
92	يوم الحساب	1294	#N/A	#N/A

Table 6.2 Sample of concrete concepts and their link to the abstract concepts

6.1.1.2 Copula Patterns

Head-modifier only extract hierarchical relation based on the structure of the multi-word terms and does not extract relations that were explicitly or inexplicitly mentioned in the text. For example, the characteristics and their categorisation such as believers, disbelievers, the guided people, and the righteous. In order to detect this kind of relations, we exploited a linguistic construct called copula. However, in some cases copula is represented by pronouns which have the information about the relation and the subject.

Figure 6.2 gives a sample of some relations found in the Qur’an text for the concept “المهتدين”, “m~uhotaduwn” which is an important concept and describes the status of a group of people in the Qur’an.



(2:16:11) *muh'tadīna*	guided-ones	فَمَا رَبِحَتْ تِجَارَتُهُمْ وَمَا كَانُوا مُهْتَدِينَ
(2:70:17) *lamuh'tadūna*	(will) surely be those who are guided	إِنَّ الْبَقَرَ تَشَابَهَ عَلَيْنَا وَإِنَّا إِنْ شَاءَ اللَّهُ لَمُهْتَدُونَ
(2:157:9) *l-muh'tadūna*	(are) the guided ones	أُولَٰئِكَ عَلَيْهِمْ صَلَوَاتٌ مِنْ رَبِّهِمْ وَرَحْمَةٌ وَأُولَٰئِكَ هُمُ الْمُهْتَدُونَ
(6:56:21) *l-muh'tadīna*	the guided-ones	قُلْ لَا أَتَّبِعُ أَهْوَاءَكُمْ قَدْ ضَلَلْتُ إِذًا وَمَا أَنَا مِنَ الْمُهْتَدِينَ
(6:82:11) *muh'tadūna*	(are) rightly guided	أُولَٰئِكَ لَهُمُ الْأَمْنُ وَهُمْ مُهْتَدُونَ
(6:117:11) *bil-muh'tadīna*	of the guided-ones	هُوَ أَعْلَمُ مَنْ يَضِلُّ عَنْ سَبِيلِهِ وَهُوَ أَعْلَمُ بِالْمُهْتَدِينَ
(6:140:20) *muh'tadīna*	guided-ones	قَدْ ضَلُّوا وَمَا كَانُوا مُهْتَدِينَ
(7:30:16) *muh'tadūna*	(are the) guided-ones	وَيَحْسَبُونَ أَنَّهُمْ مُهْتَدُونَ
(9:18:23) *l-muh'tadīna*	the guided ones	وَلَمْ يَخْشَ إِلَّا اللَّهَ فَعَسَىٰ أُولَٰئِكَ أَنْ يَكُونُوا مِنَ الْمُهْتَدِينَ
(10:45:20) *muh'tadīna*	the guided ones	قَدْ خَسِرَ الَّذِينَ كَذَّبُوا بِلِقَاءِ اللَّهِ وَمَا كَانُوا مُهْتَدِينَ
(16:125:22) *bil-muh'tadīna*	of the guided ones	إِنَّ رَبَّكَ هُوَ أَعْلَمُ بِمَنْ ضَلَّ عَنْ سَبِيلِهِ وَهُوَ أَعْلَمُ بِالْمُهْتَدِينَ
(28:56:13) *bil-muh'tadīna*	(of) the guided ones	وَلَٰكِنَّ اللَّهَ يَهْدِي مَنْ يَشَاءُ وَهُوَ أَعْلَمُ بِالْمُهْتَدِينَ
(36:21:7) *muh'tadūna*	(are) rightly guided	اتَّبِعُوا مَنْ لَا يَسْأَلُكُمْ أَجْرًا وَهُمْ مُهْتَدُونَ
(43:22:11) *muh'tadūna*	(are) guided	بَلْ قَالُوا إِنَّا وَجَدْنَا آبَاءَنَا عَلَىٰ أُمَّةٍ وَإِنَّا عَلَىٰ آثَارِهِمْ مُهْتَدُونَ
(43:37:7) *muh'tadūna*	(are) guided	وَإِنَّهُمْ لَيَصُدُّونَهُمْ عَنِ السَّبِيلِ وَيَحْسَبُونَ أَنَّهُمْ مُهْتَدُونَ
(43:49:11) *lamuh'tadūna*	(will) surely be guided	وَقَالُوا يَا أَيُّهَ السَّاحِرُ ادْعُ لَنَا رَبَّكَ بِمَا عَهِدَ عِنْدَكَ إِنَّنَا لَمُهْتَدُونَ
(68:7:11) *bil-muh'tadīna*	of the guided ones	إِنَّ رَبَّكَ هُوَ أَعْلَمُ بِمَنْ ضَلَّ عَنْ سَبِيلِهِ وَهُوَ أَعْلَمُ بِالْمُهْتَدِينَ

Figure 6.2 All occurrences of the concept "m~uhotaduwn" (the guided people) in the Qur'an

The concept of "m~uhotaduwn" is clearly linked to other concepts using copula patterns.

After we prepared the collected corpora in a relational database, the AQD as explained in Chapter 4, we applied a multiple annotation sequential patterns search for extracting the "is-a" relation. Searching for sequences was needed for extracting which terms occur beside the predefined markers of copulas. The multiple annotations were used to restrict the search for a specific pattern such as copula in the given text. We defined three markers for detecting copula in the Arabicds Qur'an text. Many examples and discussion of Arabic copula and its characteristics is in [103], [104]. The markers used for detecting copular sentences in our work are the following:

6.1.1.3 Verbal Copula (kan)

This marker of copula construct is detecting "is-a" relation among learned concepts by projecting them in the following pattern.

Listing 6.1 {[root= "kwn"][Term1][Term]}

We use the search tool described in Chapter 4 for finding patterns relations between terms. Copula is a function word that links between the subject and the predicate of a triple. “Verb to be”. In Arabic there is a set called copular verbs (“Kana wa akhwataha”). Here , we only used the lemma of the verb “kwn” for simplified the problem.

The query in Listing 6.1 used for extracting “is-a” relations between concepts that are linked with verbal copula (kwn).

Translation	triples
(Allah, isa,	(الله, isa, غفور)
(Allah,isa, All-knower)	(الله, isa, عليم)
(Allah, isa, All-Encompassing)	(الله, isa, واسع)
(Allah, isa, All-Sufficient)	(الله, isa, غنيا)
(Allah, isa, All-Hearing)	(الله, isa, سميعا)
(Allah, isa, All-Appreciative)	(الله, isa, شاكرا)
(Allah, isa, All-Mighty)	(الله, isa, عزيزا)
(Allah, isa, All-Strong)	(الله, isa,قويا)

Table 6.3 Sample of “is-a” relations extracted based on the verbal copula marker

6.1.1.4 Null Copula

Sometimes subject and predicate linked by Zero copula. This type of copula occurs in equational sentence and is also known as verbless copula [101]. In verbless copula, some sentences do not require verb or copula to link the subject and the predicate [101]. In this case, there is a link called zero copula or null copula ∅. For example, “Khaled is a doctor” is written in Arabic as “خالد دكتور”. Zero copula construct are definite nouns and their case is nominative[102]. Verbless copula also can be expressed by a pronoun. Examples from the Qur’an for this type of copula can be seen in the following:

Verse(112:2)"الله الصمد", "*Allah is the Eternal, the Absolute*". Another example is in the verse (2:179) "الحج أشهر معلومات", "*the Hajj are the months well known*". Also the verse (3:45) "المسيح عيسى", "*the Messiah Isa*".

This kind of construction can be distinguished from a complete phrase using case information. For instance, the subject in the Arabic language in most cases is nominative and the predicate is also nominative in verbless sentences [128]. Listing 6.2 shows the query used for extracting null copula in this work. As traditional Arabic grammar describes the case and features of the sequences of this construct are nominative and definite nouns, we restricted the search for all nominative terms co-occur in a window of 1. Table 6.4 gives an example of the output for this query.

Listing 6.2

```
{[Term1 & case=="nom"][Term2 & case=="nom"]}
```

Translation	Triple
(The Hajj, is-a, well known months)	(حج, is-a,اشهر معلومات)
(Allha , is-a , forgiving most merciful)	(الله, is-a, غفور رحيم)
(Allah, is-a, Allhearing and all knowing)	(الله, is-a, عليم سميع)
(Allah, is-a, All-Mighty All-wise)	(الله, is-a, حكيم عزيز)
(Allah, isa, Absolute, Clement)	(الله, is-a, حليم غنى)
Allah, isa, Mighty, Able to) Requite (the wrong)	(الله, is-a, ذو عزيز)
(Messiah, is-a, Isa)	(مسيح, is-a, ابن عيسى)

Table 6.4 Examples of triples generated based on Null Copula marker

6.1.1.5 Pronominal Copula

The copula function in Arabic may occurs in the text in form of pronouns in equational sentences [104]. Copula construct occurs as pronominal in a number of verses in the Qur'an text such as the pronominal copula that inexplicitly links between the two terms. Most occurrences of this concept show that their hyponyms are mentioned as pronouns. For example the following verse clearly explains this relations and the hyponyms of pronouns.

Verse(2:5) "أولئك هم المفلحون", "*Those are upon [right] guidance from their Lord, and it is those who are the successful*".

Another example, the verse (2:12) has a pronominal copula "هم المفسدون" "*They are the ones who spread the corrupting*".

The reference for the first example is "المتقون" (The guided people) they is and for the second example is "المنافقين" (The hypocrites) in the QurAna dataset.

It is clearly can be seen that there is an inexplicit "is-a" relations between the guided people that they are a subtype of the successful people. However, our discovery shows that most of extracted pairs related to each other as co-hyponyms rather than hyponyms as Table 6.6 and Table 6.7 show.

In copula construct one of the pair is not mentioned and instead a pronoun that refers to its concept was mentioned. This type of relations is very important for understanding the text of the Qur'an and for retrieving more relevant information about a given query. In order to extract the triples of inexplicit "is-a" relations, we adopted AQD and annotation-based search from Chapter 4 as the AQD has the pronominal information of all personal pronouns. The referent of the pronoun used as a subject of the triple and the lemma of the mentioned one used as the object. Thus, the triples of the above examples will be like (The guided people, is-a, the successful people) and (The hypocrites ,is-a, the corrupters). To extract this pattern we applied a query to find third person pronouns that are not a clitic and its antecedent in the concept table match one of our extracted concepts. We have done the query for 1st, 2nd and 3rd person, but most instances were for the 3rd person type. Let's explain this occurrences by an example from the Qur'an. There is a concept in the Qur'an known as "m~uhotaduwn" which is linked to its instances in a complex way as it can be seen in Figure 6.2. For extracting triples that their subject and predicate related to each other as hypernym-hyponym, lexico-syntactic patterns were prepared for detecting copula construct as the following:

Listing 6.3

```
{[pos!= "NEG" & pos!="P"][type=="STEM" &
person==3 & pos=="PRON"][Term]}
```

Referent	Referent	pronoun	referent	Referent	Triple
الله	Allah	هو	البر	bar~u	(الله, is-a,البر)
الله	Allah	انت	التواب	t~aw~aAbu	(الله, is-a, التواب)
الله	Allah	انا	التواب	t~aw~aAbu	(الله, is-a, التواب)
الله	Allah	هو	الحق	Haq~u	(الله, is-a, الحق)
الله	Allah	هو	الحكيم	Hakiymu	(الله, is-a, الحكيم)
الله	Allah	هو	الحي	HaY~u	(الله, is-a, الحي)
الله	Allah	هو	الخلق	xal~a`qu	(الله, is-a, الخلق)
الله	Allah	نحن	الخلق	xa`liquwna	(الله, is-a, الخلق)
الله	Allah	هو	الرحمن	r~aHoma`nu	(الله, is-a, الرحمن)
الله	Allah	هو	الرحيم	r~aHiymu	(الله, is-a, الرحيم)
الله	Allah	هو	الرزاق	r~az~aAqu	(الله, is-a, الرزاق)
الله	Allah	انت	الرقيب	r~aqiyba	(الله, is-a, الرقيب)
المنافقين	the hypocrites	هم	الخسرين	xa`siruwna	(المنافقين, is-a, الخسرين)
المنافقين	the hypocrites	هم	السفيه	s~ufahaA^'u	(المنافقين, is-a, السفيه)
المنافقين	the hypocrites	هم	الظالم	Z~a`limuwna	(المنافقين, is-a, الظالم)
المنافقين	the hypocrites	هم	العدو	Eaduw~u	(المنافقين, is-a, العدو)
المنافقين	the hypocrites	هم	الفاسق	fa`siquwna	(المنافقين, is-a, الفاسق)
الكافرين	(Kaafir) the infidels	هم	الخسرين	xa`siruwna	(الكافرين, is-a, الخسرين)
الفاسقين	disobedient	هم	الخسرين	xa`siruwna	(الفاسقين, is-a, الخاسرين)
خاسرون	loosers	هم	الخسرين	xa`siruwna	(خاسرون, is-a, الخاسرين)

مدين قوم شعيب	Madyan the people of Shuaib	هم	الخسرين	xa`siriyna	(مدين قوم شعيب, is-a, الخسرين)
حزب الشيطان	party of Satan	هم	الخسرين	xa`siruwna	(حزب الشيطان, is-a, الخسرين)

Table 6.5 Examples of triples generated based on Pronoun Copula

Let take the example above the verse (2:12).

Transformation Steps	**A triple**
An instance of the query	Humu l-muf'sidūna
Extract Pronoun Copula	(Humu, is-a, l-muf'sidūna)
Replace pronoun with Referent	(l-munāfiqīna, is-a, l-muf'sidūna)

Figure 6.3 An example of transforming text into a triple based on pronominal copula

The following pronominal copula patterns were prepared to extract inexplicit relations of "is-a" between pronoun antecedent and the term occurs next to the copula construct. Each square [] represent a segment in the prepared corpus, We search for a sequence of segments that satisfies the given conditions as attribution value pairs.

C1 Pronominal Copula {[pos=="PRON" & person ==3 & type="STEM"]}

Referent	Referent	pronoun	referent	Referent	Triple
الله	Allah	هو	البر	bar~u	(الله, is-a, البر)
الله	Allah	هو	التواب	t~aw~aAbu	(الله, is-a, التواب)
الله	Allah	انت	التواب	t~aw~aAbu	(الله, is-a, التواب)
الله	Allah	انا	التواب	t~aw~aAbu	(الله, is-a, التواب)
الله	Allah	هو	الحق	Haq~u	(الله, is-a, الحق)
الله	Allah	هو	الحكيم	Hakiymu	(الله, is-a, الحكيم)
الله	Allah	هو	الحي	HaY~u	(الله, is-a, الحي)
الله	Allah	هو	الخلق	xal~a`qu	(الله, is-a, الخلق)

الله	Allah	نحن	الخلق	xa`liquwna	(الله, is-a, الخلق)
الله	Allah	هو	الرحمن	r~aHoma`nu	(الله, is-a, الرحمن)
الله	Allah	هو	الرحيم	r~aHiymu	(الله, is-a, الرحيم)
الله	Allah	هو	الرزاق	r~az~aAqu	(الله, is-a, الرزاق)
الله	Allah	انت	الرقيب	r~aqiyba	(الله, is-a, الرقيب)
الله	Allah	نحن	الزرعون	z~a`riEuwna	(الله, is-a, الزرعون)
الله	Allah	انت	السميع	s~amiyEu	(الله, is-a, السميع)
الله	Allah	هو	السميع	s~amiyEu	(الله, is-a, السميع)
الله	Allah	انت	العزيز	Eaziyzu	(الله, is-a, العزيز)
الله	Allah	هو	العزيز	Eaziyzu	(الله, is-a, العزيز)
الله	Allah	هو	العلي	EaliY~u	(الله, is-a, العلي)
الله	Allah	انت	العليم	Ealiymu	(الله, is-a, العليم)
الله	Allah	هو	العليم	Ealiymu	(الله, is-a, العليم)
الله	Allah	هو	الغفور	gafuwru	(الله, is-a, الغفور)
الله	Allah	انا	الغفور	gafuwru	(الله, is-a, الغفور)
الله	Allah	هو	الغنى	ganiY~u	(الله, is-a, الغنى)
الله	Allah	هو	الفتاح	fat~aAHu	(الله, is-a, الفتاح)
الله	Allah	هو	القادر	qaAdiru	(الله, is-a, القادر)
الله	Allah	هو	القاهر	qaAhiru	(الله, is-a, القاهر)
الله	Allah	هو	القوى	qawiY~u	(الله, is-a, القوى)
الله	Allah	هو	اللطيف	l~aTiyfu	(الله, is-a, اللطيف)
الله	Allah	نحن	المنشئون	mun$i_#uwna	(الله, is-a, المنشئون)
الله	Allah	نحن	الوارث	wa`rivuwna	(الله, is-a, الوارث)
الله	Allah	نحن	الوارث	wa`riviyna	(الله, is-a, الوارث)
الله	Allah	هو	الوحد	wa`Hidu	(الله, is-a, الوحد)
الله	Allah	هو	الولى	waliY~u	(الله, is-a, الولى)
الله	Allah	انت	الوهاب	wah~aAbu	(الله, is-a, الوهاب)

Table 6.6 A set of “is-a” triples for the concept of “Allah”

subject	subject	pronoun	referent	Referent	Triple
المنافقين	the hypocrites	هم	الخسرين	xa`siruwna	(المنافقين, is-a, الخسرين)
المنافقين	the hypocrites	هم	السفيه	s~ufahaA^'u	(المنافقين, is-a, السفيه)
المنافقين	the hypocrites	هم	الظالم	Z~a`limuwna	(المنافقين, is-a, الظالم)
المنافقين	the hypocrites	هم	العدو	Eaduw~u	(المنافقين, is-a, العدو)
المنافقين	the hypocrites	هم	الفاسق	fa`siquwna	(المنافقين, is-a, الفاسق)
المنافقين	the hypocrites	هم	الكذب	ka`*ibuwna	(المنافقين, is-a, الكذب)
المنافقين	the hypocrites	هم	المفسد	mufosiduwna	(المنافقين, is-a, المفسد)

Table 6.7 A set of "is-a" triples occur between the concept "the hypocrites" and its sub concepts.

خسارة	loss	هو	الخسران	xusoraAnu	(خسارة, is-a, الخسران)
الكافرين	(Kaafir) the infidels	هم	الخسرين	xa`siruwna	(الكافرين, is-a, الخسرين)
الفاسقين	disobedient	هم	الخسرين	xa`siruwna	(الفاسقين, is-a, الخسرين)
خاسرون	loosers	هم	الخسرين	xa`siruwna	(خاسرون, is-a, الخسرين)
مدين قوم شعيب	Madyan the people of Shuaib	هم	الخسرين	xa`siriyna	(مدين قوم شعيب, is-a, الخسرين)
حزب الشيطان	party of Satan	هم	الخسرين	xa`siruwna	(حزب الشيطان, is-a, الخسرين)
المنافقين	the hypocrites	هم	الخسرين	xa`siruwna	(المنافقين, is-a, الخسرين)
من يضلل الله	whom Allah sends astray	هم	الخسرين	xa`siruwna	(من يضلل الله, is-a, الخسرين)

Table 6.8 A number of hyponyms belong to the class "**الخاسرين**"

6.2 Results

2287 hyponyms instances were extracted using Head-modifier, only 370 were not correct. While using copula construct extracted 364 only 23 of them were not correct.

Figure 6.4 depicts the term "القوم" (people) along with their hyponyms based on Head-Modifier and Table 6.9 shows examples of instances were extracted using copulas construct.

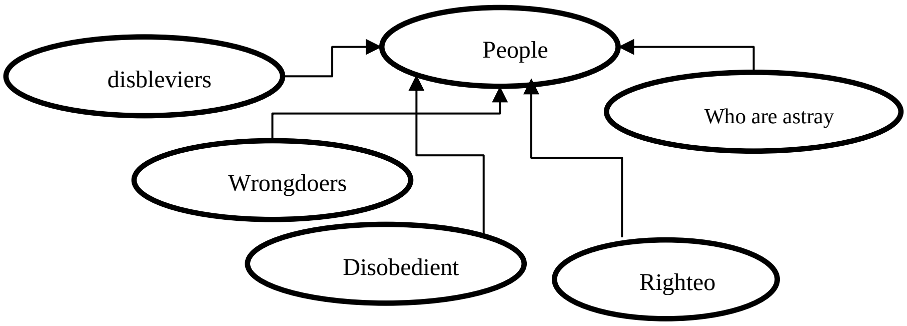


Figure 6.4 A partial view of the constructed hierarchies for the Qur'anic domain.

Copula	Verse no	Arabic	Translation	Subject	Predicate
Pronominal copula	(2:91)	وهو الحق	It is the truth	The Quran	Truth
	(2:27)	هم الخاسرون	They are the losers	Disobedient	Losers
Null copula	(20:14, 23:116)	الله الملك الحق	Allah is the true King	Allah	Truth
	(112:2)	الله الصمد	Allah is the Eternal	Allah	Eternal
Kaan	(4:92)	كان الله غنيا	Allah is All-knowing	Allah	All-knowing

Table 6.9 Examples of different types of copulas extracted from the Qur'an

A manual validation was applied to assess the quality of the extracted triples. During the validation each extracted triple was reviewed whether it makes a correct relation or not by applying binary values {0,1} 0 assigned for irrelevant relations to be filtered out and 1 for relevant ones. In order to validated the correction of retrieved relations, we calculated the precision based on equation (6.1). For example, the precision of the relations extracted by the head-modifier marker is 1917/2287 = 0.838.

$$precision = \frac{the\ number\ of\ relevant\ relations\ returned}{the\ number\ of\ relations\ founded\ by\ the\ algorithm} \qquad (6.1)$$

Is-a extraction	Precision
"is-a" based on head-modifier	0.857
"is-a" based on pronominal Copula	0.962
"is-a" based on null Copula	0.857
"is-a" based on verbal Copula	0.909
Total precision for is-a	0.936

Table 6.10 The precision of the extracted "is-a" relationships.

Table 6.10 shows the precision of extracted "is-a" relations based on a number of markers and methods. Although the number of instances was extracted using Head-Modifier is more than the instances extracted by the markers of copula construct, "is-a" relation using copula construct achieved higher precision. The Head-Modifier relies on multi-word terms, which include nested terms and not all nested terms composed of nouns. Due to not all multi-terms were headed with nominal for example, a term like "المغضوب عليهم" (Those who earned your wrath on themselves) is composed of noun + preposition + pronoun, Head-Modifier would extract incorrect instances in this case. Results show that pronoun achieved best accuracy in extracting relations between single or multi-word terms with 0.962. This type of relations

For future work, we suggest applying Head-modifier only for simple terms, which only composed of nouns.

6.3 Encoding into Ontology Formal Language

After validating generated "is-a" relations, we encoded the validated individuals and triples into OWL format using Jena OWL-API [130], which is a Java APIs for RDF and OWL representation. The APIs provide services for parsing, modifying, visualising and running inference over

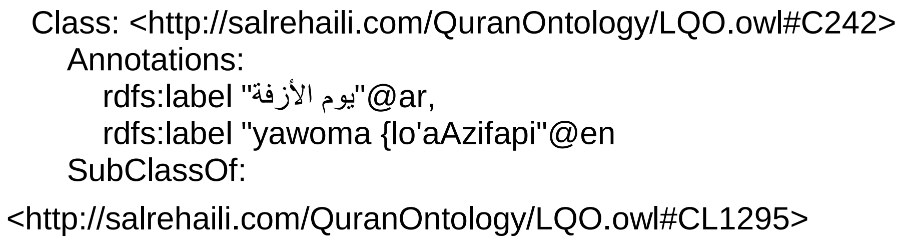
RDF and OWL. After our algorithms extracted triples, we save the valid ones in CSV files as it is shown in .

“the Day of Judgment”, is-a, “day”
“the Last Day”, is-a, “day”
“the Day of Recompense”, is-a, “day”
“the Day the account”, is-a, “day”
“known days”, is-a, “day”
“past days”, is-a, “day”
“days of misfortune”, is-a, “day”
“ the Day of Meeting”, is-a, “day”
“the Day of Calling”, is-a, “day”
“the day of the companies”, is-a, “day”
“the Day of Eternity”, is-a, “day”

Figure 6.5 an example of how CSV file is represented with triples

These triples shown above represents the one extracted using head-modifier principle. The lemma of the head terms was matched with the another terms in the identified concepts. If they are matched then the current term will be assigned as hyponeyms to the one found in the identified concepts list As it was explained in Algorithm 6.1. Then a manual validation on the extracted triples is required. After that read the triples from CSV fillies into OWL. The following example shows the previous triples encoded in OWL and linked as hyponyms to “CL1295”, which denotes the class of “day” in our ontology.

```
Class: <http://salrehaili.com/QuranOntology/LQO.owl#C293>
   Annotations:
      rdfs:label "يوم الظلة"@ar,
      rdfs:label "yawomi {lZ~ul~api"@en
   SubClassOf:
      <http://salrehaili.com/QuranOntology/LQO.owl#CL1295>
Class: <http://salrehaili.com/QuranOntology/LQO.owl#C277>
   Annotations:
      rdfs:label "يوم الخلود"@ar,
      rdfs:label "yawomu {loxuluwdi"@en
   SubClassOf:
      <http://salrehaili.com/QuranOntology/LQO.owl#CL1295>
```



```
    Class: <http://salrehaili.com/QuranOntology/LQO.owl#C242>
       Annotations:
          rdfs:label "يوم الأزفة"@ar,
          rdfs:label "yawoma {lo'aAzifapi"@en
       SubClassOf:
  <http://salrehaili.com/QuranOntology/LQO.owl#CL1295>
```

Figure 6.6 an example of how owl file represents the extracted triples

The extracted triples using copula markers encoded by the same technique described above with a little different in how to link with their hypernyms. The extracted triples were added to the parent of the term on the left side of the copula as it was explained in Algorithm 6.1.

6.4 Summary

Hierarchical relations or taxonomies are the backbone and the main components of ontologies [6], [87]–[89]. Taxonomies construction is a task of extracting pairs of terms linked by hypernym-hyponym ("is-a") relations. One of the major challenge in ontology development is automatic discovery of taxonomies. This chapter presented our methodology of building the hierarchal structure of the learned Qur'anic terms. Our methodology based on internal structural of multi-word terms headed with nominal, and three markers of copula construct. In this method, the head of a multi-word term is assuming the hypernym role [93]. Our algorithm also takes into consideration other types of terms that their heads do not make a good hypernym such as multi-word terms headed with non-nominals. Therefore, a specific linguistic construct, called copula that links the predicate to the subject in an equational sentences [104], was employed for this type of terms.

To the best of our knowledge, this is the first work which proposes a method for extracting inexplicit "is-a" relations for the Arabic language based on pronominal information. And the first work presenting a hybrid methods such as Head-Modifier and a number of Copulas patterns for extracting is-a relationships from Arabic text.

The results highlight the discovery of implicit knowledge such as the is-a relations for Allah names, and for several people categorization in the Qur'an (e.g. 'Mu'min', 'Munafiq'), etc. Results show that pronoun achieved best accuracy in extracting relations between single or multi-word terms with 0.962. As a result of this work, the valid triples were encoded into an OWL file and provided in order to be used as training set for supervised classification tasks.

Part IV
Review, Evaluation, Future Work and Conclusion

Chapter 7

Review and Evaluation

This chapter gives a critical review and evaluation on work conducted in this thesis. The aim of this thesis was to develop new methods for ontology learning from the Arabic text of the Qur'an, including concepts identification and hierarchical relationships extraction. This chapter is divided into four sections as follows.

7.1 Surveying Existing Qur'anic Resources

In order to achieve the aim of this thesis, we began with a survey, which is explained in details in Chapter 3, for reviewing a number of existing Qur'anic ontologies and annotations. The survey, which was accepted for publication in [50], covered 9 different criteria which were chosen based on some reasons explained in Section 3.1.1. This survey was needed in order to make a decision on which methodology, evaluation and other aspects should we consider. The results shown that only a few ontological annotations have been made for the entire the Qur'an. In addition, there are variations in the format used for encoding these annotations. Although we found a number of Qur'anic ontologies, some of them was made manually for only a few concepts or relations such as [56] [57], while others only cover some parts of the Quran or a limited topic such as "animals" in [17] and "praying" in [15]. Furthermore, these were not available online or even when we asked authors to send a copy of their ontologies.

7.2 The Arabic Qur'anic Database

As the background of available ontologies was reviewed, combining these ontologies based on their availability was an essential work to investigate some of linguistic features for ontological construction. This work was reported in Chapter 4 and our published paper [108]. The collections include four different types of Qur'anic annotations and not limited to ontological annotations. Some annotations were linked with the common notation of (chapter:verse:token:segment) and it was not difficult to include them. Others, like ontologies needed additional work for including them. In order to align the labels of ontologies, we conducted a hybrid-based alignment method, which is reported in our published paper [131].

We proposed an alignment approach due to the variations that occur in concept labels of ontological Qur'anic annotations. The new approach is aggregating multiple similarity scores for a given pair of concepts into a single value. Alignment was based on the aggregation scores obtained by fuzzy bilingual lexical and structure based matching of labels. Although matching based on Arabic text has achieved high precision, the aggregated scores achieved the best results as overall. In addition, annotation-based search was implemented in order to analyse the capability of extraction for ontological elements. For example, mining term candidate for the process of terms selection. The term selection is required for extracting term candidates. This combination of resources and annotation-based search offers a new environment for discovering and mining hidden Qur'anic knowledge. The syntax of the query language used in the implementation used similar syntax to Poliqrap in [114]. However as the task of extraction gets complex the query gets large and this can be seen clearly in Listing 4.6. The way the algorithm implements the query is based on the results back from the relational database system which is MySQL in our implementation. This makes it dependent on the settings of the created tables such as the relations and the types of the tables, therefore appropriate settings have to be considered to ensure an efficient response time.

7.3 Concept Identification

After a number of linguistic features were investigated, a pipeline architecture for identifying concepts from the Arabic text of the Qur'an was proposed and presented in in chapter 5. In order to improve the results of this framework it was presented along with a number of experiments.

This work is distinguished from previous research in Arabic concept identification as it uses lexical and inflectional information for term candidate extraction. In addition, it uses different schema from previous work for ranking these terms, which computes a combination of domain-specific and statistical scores for weighting and extracting the domain-specific terms. The extracted concepts of this architecture were evaluated using three evaluation metrics. The first metric was based on datasets from previous studies of the Qur'an which were used as a gold-standard. This evaluation obtained low performance due to the variation in scope and coverage they were based on. The higher coverage achieved was in comparison with QAC: 62.28%. We tried to normalise the selected dataset into the same spelling format with the extracted list, and the results have a little increase to 77.34%. The reason for that is the manually extracted concepts in these datasets do not represent the actual words that occur in the text. Partial evaluation was conducted for a selected chapter that was covered by selected datasets and volunteer annotators. The results show an agreement with hand-annotated terms and more close to them than the results collected from selected datasets. Finally, the extracted list was validated and AvP was calculated. Three experiments using AvP were conducted for improving the precision of the extracted results. The initial results were based on traditional POS patterns for term extraction. This achieved precision of up to 0.55 for the top 200 terms. The second attempt of our experiment was based on predefined patterns, it improved the precision to 0.755 for the same size of terms. The third one was based on the predefined patterns with inflection. The performance was increased to 0.905. As a result, this chapter provided a resource called AQT to be used in evaluation and as a training data-set for supervised concept identification systems. The supervised experiment was conducted using similar features as the set V3; the achieved precision was higher than V1 and V2 which gives as an indication that using case features may help in the process of candidate selection.

Finally, the chapter presented possible applications to the work presented in this thesis. An experiment of applying machine learning technique has been conducted for learning Qur'anic domain-specific terms. We exploit the resource AQT which was manually validated to build a model using machine learning. The result of this experiment compared favourably with the previous results.

7.4 Hierarchical Relationships Extraction

One of the major challenge in ontology development is hierarchical relations extraction. Our algorithm exploits a number of linguistic features such as information of internal structure of multi-word terms, the head-modifier principle for extracting the hierarchical candidates of multi-word terms. In this method, the head of a multi-word term is assuming the hypernym role [93]. Our algorithm also takes into consideration other types of terms that their heads do not make a good hypernym such as multi-word terms headed with non-nominals. Therefore, a specific linguistic construct, called copula that links the predicate to the subject in an equational sentences [104], was employed for this type of terms. In addition, the algorithm also uses the pronominal information to link the referent of a pronoun with its hypernym based on copula construct. To the best of our knowledge, this is the first work which proposes a method for extracting inexplicit "is-a" relations for the Arabic language based on pronominal information. And the first work presenting hybrid methods such as Head-Modifier and a number of Copulas patterns for extracting is-a relationships from Arabic text.

The results highlight the discovery of implicit knowledge such as the is-a relations for Allah names, and for several people categorizations in the Qur'an (e.g. 'Mu'min', 'Munafiq'), etc. Results show that pronoun achieved best accuracy in extracting relations between single or multi-word terms with 0.962 in terms of precision.

7.5 Summary

This chapter gave a critical review and evaluation on work conducted in this thesis. The aim of this thesis was to develop new methods for ontology learning from the Arabic text of the Qur'an, including concepts identification and hierarchical relationships extraction. This chapter was divided into four sections and for each section an overview of the work was giving, a critical review for the obtained results. Next chapter, outlines the thesis aims, achievements, limitations, suggestion for future work and conclusions.

Chapter 8

Future Work and Conclusions

Ontologies are very important for modelling domain knowledge among various domain fields. They provide a set of standard notations that supports formality, explicitness, sharing, reusing, they also can be queried for retrieving related knowledge and they can be reasoned with for inferring new relations. Ontologies also play an important role in different broad disciplines and their use and applications have attracted many researchers from different areas. Therefore, any attempt for automating the conversion of a textual source from a particular domain to such a model would add several benefits to the domain. The common understanding of the text, such as vocabularies, will be shared among researchers and software agents [3]. Moreover, domain knowledge can be reused and it can reuse other knowledge from other domains [3].

Different types of knowledge that exist in the Qur’an including how to deal with daily life basics such as marriage, divorce, children’s rights, parental rights, etc. Recently, a few number of attempts were conducted to model the Qur’anic knowledge using ontology. Most of previous work on the Qur’an ontology as reviewed in Chapter 3 were constructed manually or were based on encoding available annotated resources. In addition, there is lack of a well-designed methodology of ontology learning from the Classical Arabic language. The main goal of this thesis was to model the process ontology construction from the textual resources and annotations. In this thesis, we followed layer cake of [28], which outlines a number of subtasks for modelling the process on ontology construction. Here we only focused on the term extraction, concept identification and hierarchical relations acquisition. The rest of this chapter briefly summarise the whole thesis in terms of aims, achievements ,limitations and obstacles were confronted, suggestion of possible future work and conclusions.

8.1 Overview

The aim of this thesis is to develop new methods for Ontology learning from the Arabic text of the Qur'an, including concepts identification and hierarchical relationships extraction. We began with a survey for reviewing a number of existing Qur'anic ontologies against 9 different criteria. This survey was followed by combining the reviewed resources into a databased named AQD. Because of the variation in the labels of ontology, we proposed a hybrid alignment method. In addition, annotation-based search was implemented in order to analyse capability of ontological element extraction. After that, we conducted a number of experiments for identifying Qur'anic concepts. Three different arrangements of candidate selection were used with an adapted weighting scheme for ranking the most important terms. Finally, the relationships among extracted concepts were extracted based on internal information of the nominal phrases and three different types of copulas. The thesis was organised as follows:

- **Part I** included three chapters; introduction, background and the literature review.
    - **Chapter 1** introduced the thesis and give a brief background about the topic as well as discussed the motivations behind this thesis, the aim and the objectives that were listed to achieve the aim. It also highlighted contributions of the thesis and the research question that was tackled. In addition, it give a summary of the whole thesis.
    - **Chapter 2** provided background information of the related topics and methods that were mentioned such as Qur'anic domain, ontology, and some methodologies from Natural Language Processing and Machine Learning.

  - **Chapter 3** outlined and discussed relevant approaches that have been applied to similar work, focused on the domain of the Arabic text of the Qur'an, ontology learning from textual resources with great focus on concept and taxonomic relation extraction.
- **Part II** includes chapter no 4.
  - **Chapter 4** This chapter produced a multi-dimensional resources named, Arabic Qur'anic Database (AQD) for combining 5 different ontologies and annotations of the Qur'an. In addition it provided a regular expression like query for extracting complex grammatical and ontological instances from the AQD.
- **Part III** includes two chapters; 5 and 6.
  - **Chapter 5** provided a framework for generating terms-like and using weighing schema based on multiple information for measuring their relevance. In addition, a number of experiments based on the method of terms extraction and the method of evaluation were presented. An experiment on applying machine learning algorithms has been conducted for learning Qur'anic domain-specific terms. In this chapter, we exploited the resource AQT which was manually validated to build a model.
  - **Chapter 6** introduced a new approach for hierarchical relations from Arabic text of the Qur'an. A set of hierarchical relations occurring between identified concepts were extracted based on hybrid methods including head-modifier, set of rules for copula construct in Arabic, referents and derivational information.
- **Part IV** includes two chapters; 7 and 8.
  - Error! Reference source not found. presented a critical review and evaluation for the presented work in this thesis.

8.2 Findings

For other languages like Czech language, [84] reported that case feature gives better results when they used with POS. A similar results for Arabic reported in [85] for using case in gold experimental. Another work who claims that morphology is essential syntactic modelling is in

[83]. We found the morphological case feature is useful for distinguish valid multi-word terms candidates and gives the best result comparing to methods that only include POS information. The initial results was based on traditional POS patterns for term extraction. This achieved precision of up to 0.55 for the top 200 terms. The second attempt of our experiment was based on a predefined patterns, it improved the precision to 0.755 for the same size of terms. The third one was based on the predefined with inflectional. The performance was increased to 0.905.

In this thesis we also highlight the discovery of implicit knowledge such as the is-a relations for Allah names, and for several people categorization in the Qur'an (e.g. 'Mu'min', 'Munafiq'), etc. Results show that pronoun achieved best accuracy in extracting relations between single or multi-word terms with 0.962. This thesis also found the hybrid method (HB) has outperforms the structure-based (SB), fuzzy lexical-based for English (Lex_{EN}) and fuzzy lexical-based for Arabic (Lex_{AR}). Significantly, Lex_{AR} has obtained similar results but it decreases sharply after number 40 in the ranked list. Over all the new algorithm HB has outperforms the three compared algorithms.

8.3 Achieved Contributions

- A new Qur'anic database, AQD, that combines five different ontological and other annotations for the Qur'an. This resource is attached to an annotation-based search, which allows users to extract knowledge based on a query that can contain several types of features. Such an environment will enable researchers in Arabic computational linguistics and Qur'anic research to investigate new directions and new features for different NLP learning tasks.

- A new framework for Domain-Specific Terms of the Qur'an and concepts.

The proposed framework takes into consideration the importance of morphological features that detect the lexical stable unit. In addition, it combines domain-specific knowledge and statistical knowledge for measuring the importance of a term. To our knowledge, this work is the first that exploits inflectional, domain-specific knowledge and statistical knowledge in concept identification for the ontology learning task. A new standard resource for research on Classical Arabic term extraction is provided as a dataset which contains 10351 domain-specific terms validated manually. We named this dataset as Arabic Qur'anic Terms (AQT).

- An original approach of automatic hierarchical relations construction based on a combination of head-modifier and a three markers for a linguistic construct called copula. We show its success in constructing the hierarchy between identified concepts for some domains such as names of Allah. To the best of our knowledge, this is the first work proposes a method for extracting inexplicit "is-a" relations for the Arabic language based on pronominal information. And presenting a hybrid methods combine Head-Modifier and a number of Copulas patterns for extracting is-a relationships from Arabic text. We also encoded three previous Qur'anic ontological annotations (Qurany, QurAna and QAC) using Web Ontology Language (OWL). We also have done an experiment on aligning highly variant Qur'anic ontological resources.

- A new hybrid ontology alignment which is aggregating multiple similarity measures for a given pair of concept. It takes advantage of combining fuzzy bilingual lexical and structure based methods.

- A supervised model for learning implemented in an experiment for Arabic Qur'an term extraction base on AQT dataset.

8.4 Future Work

This research laid out a foundation for a number of possible future works on the ontology learning from the Arabic language. Here we categorised the possible future works into a number of points as follows:

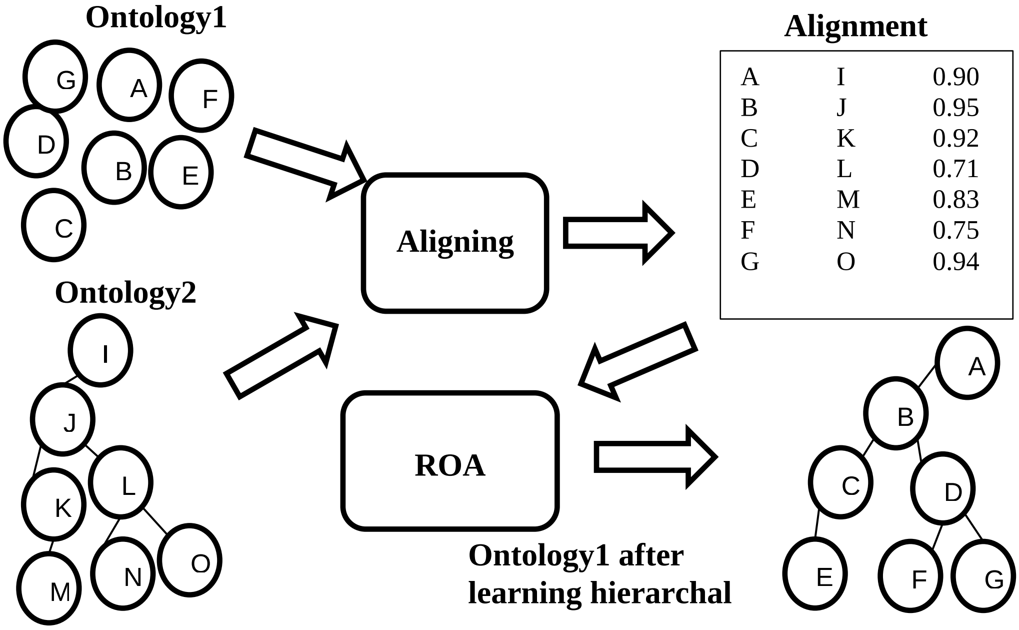
8.4.1 Concept Identification

The Arabic Qur'anic Terms (AQT), which is an output of Chapter 5, can be used as a gold-standard resources for evaluating new methods for terms extraction and concept identification. This resource is also can be useful for developing new machine learning methods as it is shown in Section 5.3. Derived terms is an important constitute of Arabic terms as it is explained in Subsection 2.6.5. A possible future work is to consider this type of terms in the process of candidate generation of multi-word terms. Another possible future work could be done using the framework presented in Chapter 5 on the text of Hadith with same methods used for generating the set V3 as it achieved the best results.

8.4.2 Hierarchical Construction

Relation over Alignment(ROA) is presented in this thesis as a proposed methodology for constructing hierarchy structure based on external ontologies from related domains. This method can be applied for any type of relations not only hierarchy as well as another type of properties for structure-based similarity can be compared. We only focused on the parental and children nodes of concepts that we want to obtain their relations.

Future work could be aimed to find the relations between corresponding terms from external resources to enrich the hierarchal structure of our ontology. Both term x_i *and* y_i are representing the same meaning if their weight is high. After many experiments we found 0.6 gives the best accuracy when using hybrid methods on formula (4.1) in Chapter 4. This idea is visualised in the Figure 8.1.



A	I	0.90
B	J	0.95
C	K	0.92
D	L	0.71
E	M	0.83
F	N	0.75
G	O	0.94

Figure 8.1 extracting relations based on based on alignment

For finding relations between ontology terms, we assume that a term is related to another term in the list if its corresponding term is related to the corresponding the term to that term. By looking for relations occur between their corresponding terms in another ontology. Let considered we have two tuples of corresponding terms as the following.

$$(x_a, y_b, 0.9)$$

$$(x_f, y_g, 0.72)$$

If there is a relation between the $y_a\ and\ y_b$ then we can infer that the two terms $x_f\ and\ x_g$ in the other ontology are related with the same relation. However, the suggestion is only to obtain "isa" relations.

8.4.3 The Arabic Qur'anic Database

This work combines different annotations for the annotation-based search of the Qur'an and it allows users to make a complex query that integrates a number of annotations in one single query. This will allow interested researcher to investigate a number of extraction tasks such as the selection of candidate terms or opposite relationships between nominals. In addition, it supports context-free grammar for complex extraction tasks. There is much room for further work here such as complex linguistic phenomenon such as dependency relationships among nominals. We plan to investigate the possibility of adding more annotations and features for the AQD. We gave an example of how to find antonyms from prepositional phrases in subsection 4.4.3. Another pattern-based ontological annotations can be discovered as AQD support different types of annotations. The tool used in the experiments is available on our website[1], together with the source code of the annotation-based search. One of the planned future work is to make it web-based application.

For the Future work of ontologies alignment, we plan to look at improving the structure-based by computing not only the children of the concepts but add more information such as parents neighbours. The returned aligned list from the new approach can be used as a training data for machine learning tasks as each pair was classified whether is correct or not. In addition, this approach can be reused for other domains that their entities tend to be expressed in many different way. We think that any computational effort on understanding or learning the Qur’anic text will be benefit to billions of Muslims and non-Muslims around the world.

8.5 Conclusion

The main contribution of this work was new methods for Ontology learning from the Arabic text of the Qur’an, including concepts identification and hierarchical relationships extraction. Ontology learning work in Arabic is less comparing to work in other languages. This thesis presented an original methodology of learning ontology elements from the Arabic text of the Qur’an which is also can be applied for any Classical Arabic text. Only few researchers have paid attention to the area of Automatic Concept Identification from Arabic text of the Qur’an.

[1] http://salrehaili.com/AQD

Concepts extraction is required as a first step for many NLP applications and this may increase the impact of this research. The set of concepts was identified based on an original pipelined architecture. The produced list of terms was validated based on three types of evaluation metrics: (i) an existing datasets, (ii) partial validation for a certain parts of the Qur'an and (iii) a common evaluation metric for ranked list called Average Precision.

One of the major challenge in ontology development is automatic discovery of taxonomies. Our algorithm for taxonomies exploits information of internal structure of multi-word terms, the head-modifier principle for extracting the hierarchical candidates of multi-word terms. In this method, the head of a multi-word term is assuming the hypernym role [93]. Our algorithm also takes into consideration other types of terms that their heads do not make a good hypernym such as multi-word terms headed with non-nominals. Therefore, a specific linguistic construct, called copula that links the predicate to the subject in an equational sentences [104], was employed for this type of terms.

Appendix A

Resources Features

A.1 The annotation set that can be used for query for extracting antonyms in preposition phrases

- Traditional Features (**TRAD**)
- Lexical Features (**LEX**)
    - Lemma
    - Root
    - Buckwalter
    - Diacritics
    -
- Syntactic Features (**SYN**)
    - Num NPs per verse
    - Number of VPs per verse
    - Number of PPs per verse
- Morphological Features (**MORPH**)
    - Derivational Morphology of the Noun (**DERIV**)
        - Active participle
        - form
- Inflectional Morphology Inflectional features deal with variation of the word forms.
    - For verbs and nominals it also called (**phi-features**)
        - Person (1st, 2nd and 3rd)
        - Gender (masculine and feminine)
        - Number (singular, dual and plural)
    - For verbs (**INFLV**)

        - ASPECT
        - VOICE
        - MOOD
    - For nominals (**INFLN**)
        - Grammatical case (nominative, accusative and genitive)
            -
        - State
        - DET

Appendix B

Terms and Concepts

B.1 Syntactic Patterns found in the discovery stage

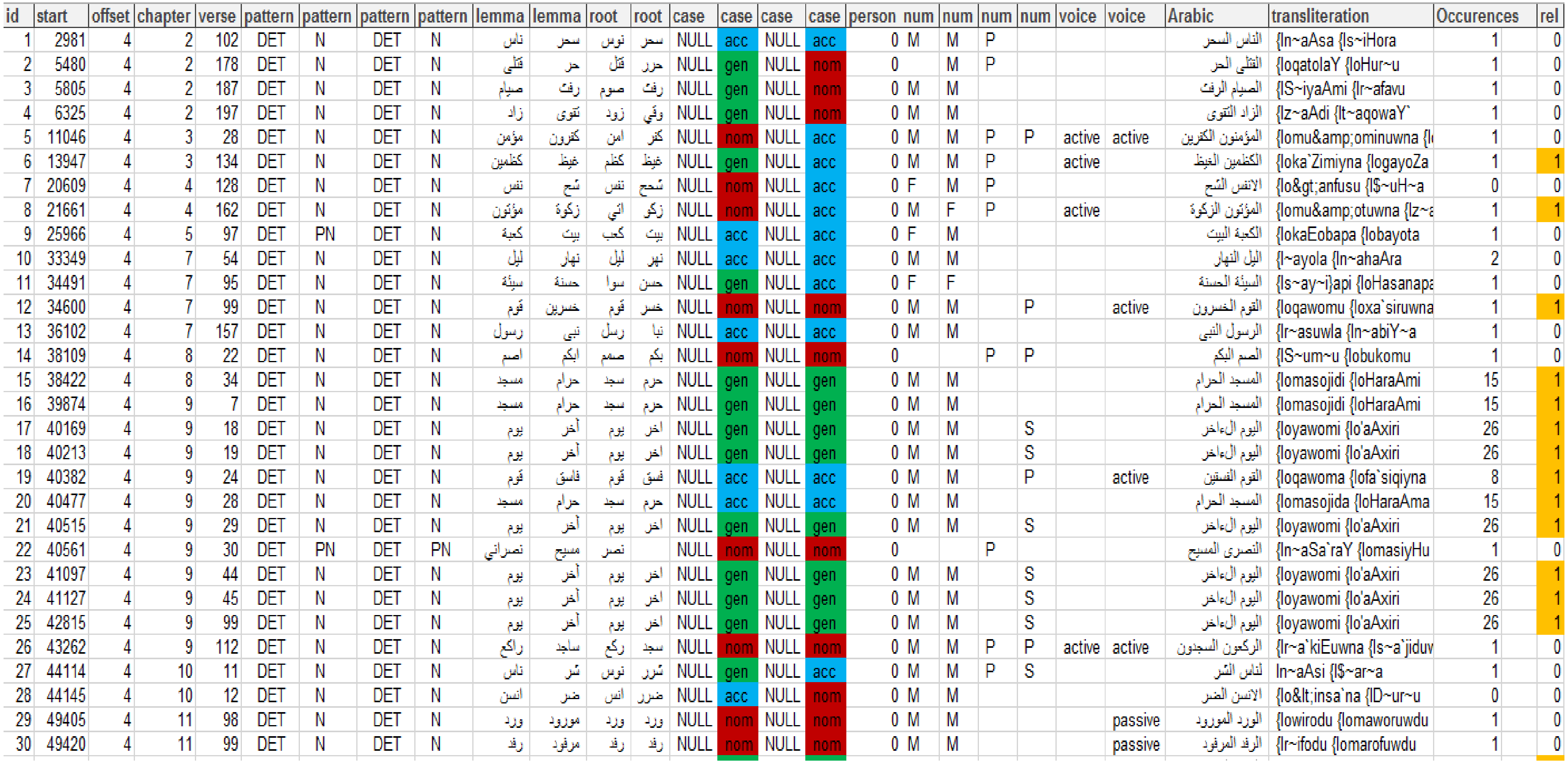
No	Syntactic Patterns	Occurrences
1	N	25136
3	DET.N	7488
4	PN	3911
6	REL V	2919
2	N N	2621
8	ADJ	1961
9	REL V.PRON	1886
7	N V.PRON	1377
5	N DET.N	1328
10	N ADJ	1049
11	N PN	783
13	DET.ADJ	575
14	DET.N DET.ADJ	420
12	N N N	334
15	DET.PN	289
16	REL V.PRON V.PRON	219
17	REL V.PRON P.PRON	199
19	REL V P.PRON	190
20	DET.N DET.N	127
21	REL V.PRON DET.N	109
22	REL V.PRON.PRON	100
25	REL V.PRON P N.PRON	98
24	REL P N.PRON	86
35	REL V.PRON P.DET.N	69
29	REL V.PRON P N PN	53
37	REL V.PRON P.N PN	53
18	REL N	52
26	REL V.PRON CONJ.V.PRON DET.N	51
30	REL V P N.PRON	49
23	N REL V.PRON V.PRON	48
27	REL V.PRON P.PRON N	45
28	REL P.PRON	45
31	REL PRON N	35

33	REL V DET.N CONJ.DET.N	32
34	REL P DET.N CONJ.DET.N	24
32	N PN ADJ	15
39	REL V P DET.N N	11
38	REL V P.PRON P.PRON N	9
41	REL V.PRON.PRON DET.N	9
40	REL V P PN N	8
36	REL V.PRON P N PN N	7
42	REL V V.PRON	7
43	REL V.PRON P.PRON DET.N	2
44	REL P N DET.N	1
45	REL V V N PN	1

Table B.1 All syntactic patterns found in the chapter 29 and their occurrences in the Qur'an based on syntactic patterns discovered above.

B.2 Case-based feature for unithood of MWTs

The case feature of MWTs constituents have been combined with the manual validated terms and a set of patterns based on the case feature was extracted for detecting lexical stable unit from not valid ones.



id	start	offset	chapter	verse	pattern	pattern	pattern	pattern	lemma	lemma	root	root	case	case	case	case	person	num	num	num	num	voice	voice	Arabic	transliteration	Occurences	rel
1	2981	4	2	102	DET	N	DET	N	ناس	سحر	نوس	سحر	NULL	acc	NULL	acc	0	M	M	P				الناس السحر	{In~aAsa {Is~iHora	1	0
2	5480	4	2	178	DET	N	DET	N	قتلى	حر	قتل	حرر	NULL	gen	NULL	nom	0		M	P				القتلى الحر	{loqatolaY {loHur~u	1	0
3	5805	4	2	187	DET	N	DET	N	صيام	رفث	صوم	رفث	NULL	gen	NULL	nom	0	M	M					الصيام الرفث	{IS~iyaAmi {Ir~afavu	1	0
4	6325	4	2	197	DET	N	DET	N	زاد	تقوى	زود	وقي	NULL	gen	NULL	nom	0	M	M					الزاد التقوى	{Iz~aAdi {It~aqowaY`	1	0
5	11046	4	3	28	DET	N	DET	N	مؤمن	كفرون	امن	كفر	NULL	nom	NULL	acc	0	M	M	P	P	active	active	المؤمنون الكفرين	{lomu&amp;ominuwna {l	1	0
6	13947	4	3	134	DET	N	DET	N	كظمين	غيظ	كظم	غيظ	NULL	gen	NULL	acc	0	M	M	P		active		الكظمين الغيظ	{loka`Zimiyna {logayoZa	1	1
7	20609	4	4	128	DET	N	DET	N	نفس	شح	نفس	شحح	NULL	nom	NULL	acc	0	F	M	P				الانفس الشح	{lo>anfusu {I$~uH~a	0	0
8	21661	4	4	162	DET	N	DET	N	مؤتون	زكوة	اتي	زكو	NULL	nom	NULL	acc	0	M	F	P		active		المؤتون الزكوة	{lomu&amp;otuwna {Iz~a	1	1
9	25966	4	5	97	DET	PN	DET	N	كعبة	بيت	كعب	بيت	NULL	acc	NULL	acc	0	F	M					الكعبة البيت	{lokaEobapa {lobayota	1	0
10	33349	4	7	54	DET	N	DET	N	ليل	نهار	ليل	نهر	NULL	acc	NULL	acc	0	M	M					اليل النهار	{I~ayola {In~ahaAra	2	0
11	34491	4	7	95	DET	N	DET	N	سيئة	حسنة	سوا	حسن	NULL	gen	NULL	acc	0	F	F					السيئة الحسنة	{Is~ay~i}api {loHasanapa	1	0
12	34600	4	7	99	DET	N	DET	N	قوم	خسرين	قوم	خسر	NULL	nom	NULL	nom	0	M	M		P		active	القوم الخسرون	{loqawomu {loxa`siruwna	1	1
13	36102	4	7	157	DET	N	DET	N	رسول	نبي	رسل	نبا	NULL	acc	NULL	acc	0	M	M					الرسول النبي	{Ir~asuwla {In~abiY~a	1	0
14	38109	4	8	22	DET	N	DET	N	اصم	ابكم	صمم	بكم	NULL	nom	NULL	nom	0			P	P			الصم البكم	{IS~um~u {lobukomu	1	0
15	38422	4	8	34	DET	N	DET	N	مسجد	حرام	سجد	حرم	NULL	gen	NULL	gen	0	M	M					المسجد الحرام	{lomasojidi {loHaraAmi	15	1
16	39874	4	9	7	DET	N	DET	N	مسجد	حرام	سجد	حرم	NULL	gen	NULL	gen	0	M	M					المسجد الحرام	{lomasojidi {loHaraAmi	15	1
17	40169	4	9	18	DET	N	DET	N	يوم	آخر	يوم	اخر	NULL	gen	NULL	gen	0	M	M		S			اليوم الءاخر	{loyawomi {lo'aAxiri	26	1
18	40213	4	9	19	DET	N	DET	N	يوم	آخر	يوم	اخر	NULL	gen	NULL	gen	0	M	M		S			اليوم الءاخر	{loyawomi {lo'aAxiri	26	1
19	40382	4	9	24	DET	N	DET	N	قوم	فاسق	قوم	فسق	NULL	acc	NULL	acc	0	M	M		P		active	القوم الفسقين	{loqawoma {lofa`siqiyna	8	1
20	40477	4	9	28	DET	N	DET	N	مسجد	حرام	سجد	حرم	NULL	acc	NULL	acc	0	M	M					المسجد الحرام	{lomasojida {loHaraAma	15	1
21	40515	4	9	29	DET	N	DET	N	يوم	آخر	يوم	اخر	NULL	gen	NULL	gen	0	M	M		S			اليوم الءاخر	{loyawomi {lo'aAxiri	26	1
22	40561	4	9	30	DET	PN	DET	PN	نصراني	مسيح	نصر		NULL	nom	NULL	nom	0			P				النصرى المسيح	{In~aSa`raY {lomasiyHu	1	0
23	41097	4	9	44	DET	N	DET	N	يوم	آخر	يوم	اخر	NULL	gen	NULL	gen	0	M	M		S			اليوم الءاخر	{loyawomi {lo'aAxiri	26	1
24	41127	4	9	45	DET	N	DET	N	يوم	آخر	يوم	اخر	NULL	gen	NULL	gen	0	M	M		S			اليوم الءاخر	{loyawomi {lo'aAxiri	26	1
25	42815	4	9	99	DET	N	DET	N	يوم	آخر	يوم	اخر	NULL	gen	NULL	gen	0	M	M		S			اليوم الءاخر	{loyawomi {lo'aAxiri	26	1
26	43262	4	9	112	DET	N	DET	N	راكع	ساجد	ركع	سجد	NULL	nom	NULL	nom	0	M	M	P	P	active	active	الركعون السجدون	{Ir~a`kiEuwna {Is~a`jiduw	1	0
27	44114	4	10	11	DET	N	DET	N	ناس	شر	نوس	شرر	NULL	gen	NULL	acc	0	M	M	P	S			لناس الشر	In~aAsi {I$~ar~a	1	0
28	44145	4	10	12	DET	N	DET	N	انسن	ضر	انس	ضرر	NULL	acc	NULL	nom	0	M	M					الانسن الضر	{lo<insa`na {ID~ur~u	0	0
29	49405	4	11	98	DET	N	DET	N	ورد	مورود	ورد	ورد	NULL	nom	NULL	nom	0	M	M				passive	الورد المورود	{lowirodu {lomaworuwdu	1	0
30	49420	4	11	99	DET	N	DET	N	رفد	مرفود	رفد	رفد	NULL	nom	NULL	nom	0	M	M				passive	الرفد المرفود	{Ir~ifodu {lomarofuwdu	1	0

Table B.2 Morphological Case feature analysis for MWTs

B.3 Further Reading on Related Work of Arabic Term Extraction

References	Purpose	Methodology	Results	Term types	Domain	Strengths	Weaknesses
[37]	multi-word term extraction.	They started with defining the linguistic specification of MWTs for Arabic language. Then they select MWT based on N1 ADJ| N1 P? N2.Then they evaluated several statistical measures. Such as LLR, FLR, MI3 and t-scores. FLR and t-score have shown significant results.	Compared to a list of known Arabic terms from the same domain. Each extracted term was given scores for the selected statistical measures. Then ranked these terms and take 100 and compare it to the reference list. Only precision was computed where recall is not. FLR	Bi-gram of two nouns and a noun flowed by adjective. And tri-gram for noun followed by preposition followed by noun.	newspaper articles for environment domain		

			measure was the highest with 60%.			
[79] [40]	A general purpose bilingual (English/Arabic) keyphrases extraction system.	Candidate phrases selected using a set of rules; (1)phrases cannot separated by punctuation marks or stop words (2) a phrase has to have appeared at least n times. The least allowable seen frequency (lasf) factor. (4) the position CutOff . Then TF-IDF used for weighting candidate kephrases. After that certain number of keyphrases is generated for what user asked.	For Arabic, 100 documents was collected from Arabic Wikipedia and Agricultural thesaurus, then the results were compared to	Scientific articles From agricultural domain	No need for training data	It requires an understanding of the targeted text in order to configure Like s list of stop words for the domain. Qur'an has different stopwords The lasf factor should be assigned carefully and it depends on the size of the selected document and their language as the author were chosen different values for Arabic and English because of the

							specification of Arabic phrases occurrences in a document. TF-IDF is not suitable for low-frequent data
[77]	A topic modelling.	A package of R language, which is called, was used for building Document Term matrix (DTM) and for creating LDA model. Three DTMs of words, roots and stems were created using the frequencies and TF-IDF.	LDA technique showed best results when applying on verses. And Log-Likelihood	Single-word terms	The Qur’an for the chapter of Joseph		
[76]	Simple-term and collocations extraction approach for building an ontology for the Qur’an	A pure statistical method (TF-IDF) used for extracting simple-terms. TF-IDF was used for weighting simple-terms then sorted them in descending order and pick those are above the threshold.	Simple-terms achieved 0.88 precision, 0.92 recall and 0.89 f-measure. Extracting collocations	Simple-terms and compound of two words	The Qur’an		Some verbs, prepositions and attached to the extracted terms such as “بسم”, “لوقعتها”, “باكواب”.

		The paper was not give the threshold used. Then JAPE used for extracting collocations of Noun-Adjective, Noun-Noun and Verb-Noun.	achieved 0.5 precision, 1 recall and 0.66 f-measure. After calculating mutual information for collocations, the precision improved to be 0.86		Among collocations, there are “الجاهل اغنياء”
[7]	Association rules for Qur’anic ontology construction	Pattern-based used for extracting concspts and association rules for the realtions Applied KP-Miner for keyphrase extraction, did not		12 chapters consist of stories and prophet names Prophetic stories	Qur’anic terms are various in different form and this proposed method will only cover few instances from the Qur’an. Especially with multi-word terms the variations becomes more and more,

		Then stemming the extracted phrases using a stemmer from () Then association rules used for extracting relationships among these phrases			
[70]	Proposed a MWT system based on contextual information	Start with linguistic approach Taani's POS tagger for extracting candidates with the patterns of Noun- Prep-Noun and Noun-Noun. Then statistical approach based on Termhood and Unithood was applied for weighting the candidates	They evaluated based on comparision to other association measure methods NTC-Value, NC-Value and C-Value. In term of precision and deal with tri-gram terms	environment domain	Only precision was mentioned and ignore recall.

[80]	Methods for extracting keyword and keyphrase candidate

Table B.3 Related work on Arabic term extraction

B.4 The Code of multi-word terms selection

```
import sys
import nltk
from nltk.tree import Tree
import pymysql
import re
from collections import Counter
import math

sys.setrecursionlimit(200000)


with open('quran/quran-uthmani-min_partly_dict.txt', 'r') as myfile1:
        domain=myfile1.read().replace('\n', '')

with open('quran/gloss2.txt', 'r') as myfile2:
        gloss=myfile2.read().replace('\n', '')

maxDomainTerm = 2670
maxGlossTerm = 87

def getTermfrequency(term, target):
    return target.count(term)

def headtail(term):
    head, space, tail = term.partition(' ')
    return head, tail


def occurenceees(term,d):
    #print ("count", getfrequency(term, domain))
    #print("counter", d[term])
    #return d[term]
    return getTermfrequency(term, d)


def tf(term):
    #thisprobability from Kang's paper (Kang,2014)
    #p=occurenceees(term,C)/MaxOccurences(SC)
    p=occurenceees(term,domain)/maxDomainTerm
    return p

def Wd(term):
    #thisprobability from Kang's paper (Kang,2014)
    #p=1+(math.log10(occurenceees(term,C))/math.log10(MaxOccurences(SC)))
    p=1+(math.log10(occurenceees(term,gloss)+1)/math.log10(maxGlossTerm))
    return p

def W(term):
    # This measures the dgree of a term t is related to a given domain-specific
    # Multi-level termhood
    # Termhood sometimes computed based on frequecny and rank difference between
domain and general corpus
    stopwords=file2list("quran/stopwords.txt")
    f=[x for x in stopwords if term == x]

    if len(term.split())==1:
        if len(f)>=1:
            return 0
        else:
            return tf(term) * Wd(term)
    else:
        #co_occurrence=L.count(term)
        head, tail = headtail(term)
        return (tf(head)* Wd(head))+W(tail)

#------------------------------------------------------------------------------------------------------------------------

def loadtaggedsenence(filename):
    tagged_sentences=list()
    with open(filename, encoding='utf-8') as file:
        for line in file:
            tagged_sentences.append([nltk.tag.str2tuple(t) for t in line.split()])
    file.close()
    return tagged_sentences

def tagged_sentence2tuple(tagged):
    tagged_sentences=[nltk.tag.str2tuple(t) for t in tagged.split()]
    return tagged_sentences

def tuple2sentence(tup):
    str1=""
    for i in tup:
        str1+=" "+nltk.tag.tuple2str(i)
    return str1

def leaves(self):
        """
        Return the leaves of the tree.

            >>> t = Tree.fromstring("(S (NP (D the) (N dog)) (VP (V chased) (NP (D the) (N
cat))))")
            >>> t.leaves()
            ['the', 'dog', 'chased', 'the', 'cat']

        :return: a list containing this tree's leaves.
            The order reflects the order of the
            leaves in the tree's hierarchical structure.
        :rtype: list
        """
        leaves = []
        for child in self:
            if isinstance(child, Tree):
                leaves.extend(child.leaves())
            else:
                leaves.append(child)
        return leaves
#for idx, i in enumerate(tagged_sentences):
    #print(idx+1,i)
#tagged_token = nltk.tag.str2tuple('fly/NN')
#{(<DET>?<N>+)*<DET>?<PN>}
#NP: {(<DET>?<N>+)+((<DET>?<ADJ>+)|<DET>?<PN>)*}

def NP_Chunker(tagged_sentences, grammar):
    tree = list()
    cp = nltk.RegexpParser(grammar)
    for idx,sent in enumerate(tagged_sentences):
        tree.append(cp.parse(sent))
        #print(tree)
    #print("result is ",tree)
    return tree

def list2file(l, fname, op):
    fo = open(fname, op)
    for item in l:
        fo.write(str(item).strip() + "\n")
    fo.close()
    #pd = pandas.DataFrame(l)
    #pd.to_csv(fname, mode = 'a', sep='\t', encoding='utf-8')

def getquery(cur):
    #cur.execute(sql)
    L=list()
    for row in cur:
        L.append(row)
    return L
def file2list(fname):
    return list(open(fname,'r'))

def getWord(wor_id, line):
    txt1 = file2list("quran/quran-simple-clean.txt")
    txt2 = file2list("quran/quran-simple-enhanced.txt")
    txt3 = file2list("quran/quran-uthmani.txt")


    word1 = txt1[line-1].split()[wor_id-1]
    word2 = txt2[line-1].split()[wor_id-1]
    word3 = txt3[line-1].split()[wor_id-1]
    return word1, word2, word3

with open('quran/quran-uthmani-min_partly_dict.txt', 'r') as myfile:
        data=myfile.read().replace('\n', '')

def getfrequency(term):
    return data.count(term)

def get_concated_sentences(sidSentence_in_tup):
    conn=pymysql.connect(host="localhost",user="root", passwd="982",db="Qurandb",
charset="utf8")
    cursor = conn.cursor()
    strtmp=""
    transletration=""
    lemmaD = ""
    rootD=""
    case1=""
    pos=""
    person=""
    gender=""
    num=""
    voice=""
    for i in sidSentence_in_tup:
        sql = "SELECT diacritics, type, ch_id, ver_id, buckwalter, lemmaD, rootD, case1, pos,
person, gender, num, voice from segments WHERE sid = " + i[0]
        cursor.execute(sql)
        Terms=getquery(cursor)
        ch=Terms[0][2]
        ver=Terms[0][3]
        lemmaD=lemmaD +", "+ Terms[0][5].strip()
        rootD=rootD+ ", "+ Terms[0][6].strip()
        case1=case1+", "+Terms[0][7].strip()
        pos=pos+", "+Terms[0][8]
        person=person+", "+str(Terms[0][9])
        gender=gender+", "+Terms[0][10]
        num=num+", "+Terms[0][11]
        voice=voice+", "+Terms[0][12]
        if Terms[0][1].strip()=="SUFFIX":
            #remove previous white space
            strtmp=strtmp[0:len(strtmp)-1]
            transletration=transletration[0:len(transletration)-1]
            #strtmp=strtmp.strip()
            strtmp+=Terms[0][0]+" "
            transletration+=Terms[0][4]+" "
            #print(strtmp)
        elif Terms[0][1].strip()=="PREFIX":
            strtmp += Terms[0][0].strip()
            transletration+=Terms[0][4].strip()
        else:
            strtmp += Terms[0][0].strip()+" "
            transletration+= Terms[0][4].strip()+" "
    cursor.close()
    conn.close()

    return strtmp, ch, ver, transletration, lemmaD, rootD, case1, pos, person, gender,num,
voice, getfrequency(strtmp.strip()), W(strtmp.strip())

def parsing(grammar, tagged_sentences_file):
    tagged_sentences=loadtaggedsenence(tagged_sentences_file)  # load tagged sentence
file

    a='''
    #NP0:{<UN>*<DN>*}
    #NP2: {<REL>(<V>?<P>?<PRON>?<P>?<DN>?)*}
    #NP3:{<NP1>?<NP2>?<PRON>?}          # DADJ -> DET ADJ Defined Adjectives
    #NP3:{<P><UN><UN><UN>}
    #NP4:{<NP1><P><NP1>}
    #NP1:{<UN>*<UADJ>?<DN>?<DADJ>*}
    #NP:{<UN>*<A>}'''
    tree = NP_Chunker(tagged_sentences, grammar)        # chunking
    tmp = list()
    concated= list()
    for s,n in enumerate(tagged_sentences):
        tmp.append('')
        concated.append('')
        for subtree in tree[s].subtrees():
            if subtree.label() ==  'NP':
                #tmp.append(tuple2sentence(leaves(subtree))) # return it to the original tagged
sentence
                tmp[s]=tuple2sentence(leaves(subtree))


                offsetsid=tmp[s].strip().split(" ")
                start=offsetsid[0].split("/")[0]
                offset = len(offsetsid)

                #print(tuple2sentence(leaves(subtree)), start, offset)

                #concated.append(get_concated_sentences(leaves(subtree)))
                strtmp, ch, ver, transletration, lemmaD, rootD, case1, pos, person, gender, num,
voice, frequency, W  = get_concated_sentences(leaves(subtree))
```

```
                concated[s]=str(start)+ ", " +str(offset) +", "+ str(ch) + "," + str(ver) + ", " + pos + ", "
+lemmaD + ", "  + rootD +", " +case1 +", "+ person +", " + gender +", " +num + ", " + voice +",
"+ strtmp + ", " + transletration + ", "  + str(frequency) +", "+ str(W)
                #print(subtree)
    return tmp, concated
def mergeLists(l1,l2):
    for i, l in enumerate(l1):
        l1[i]=l1[i] + ', ' +l2[i]
    return l1

def caseFilter(tmp, order1, case_values):
    cClass=['false']* len(tmp)
    conn=pymysql.connect(host="localhost",user="root", passwd="982",db="Qurandb",
charset="utf8")
    cursor = conn.cursor()
    print(order1)
    for i in range(len(tmp)):
        units_candidates= re.split('/| ', tmp[i].strip())
        #print (units_candidates)
        for ib, bb in enumerate(case_values):   # for idx, o in enumerate(order1)
            flag=""
            for idx, o in enumerate(order1):   #for ib, bb in enumerate(case_values):
                if(len(units_candidates)>1):
                    case1=bb.split(' ')
                    sql = "SELECT trim(case1) from segments WHERE sid = " + units_candidates[o]
                    #print(sql)
                    cursor.execute(sql)
                    ccc= getquery(cursor)[0]
                    if ccc[0]==case1[idx] and cClass[i]=='false':
                        #print(order1[idx], ccc[0], case1[idx])
                        flag=flag + case1[idx]+" "
            if(flag.strip()==bb):
                cClass[i]='true'
#    for i in tmp:
#        units_candidates= re.split('/| ', i.strip())
#        for idx, o in enumerate(order):
#            case1=case_values.split(' ')
#            sql = "SELECT case1 from segments WHERE sid = " + units_candidates[o]
#            cursor.execute(sql)
#            if getquery(cursor)[0].split()==case1[idx] and cClass[i]=='false':
#                cClass[i]=true
#            else:
#                cClass[i]=false
#                break
    return cClass

print("----------------------------------------------------------------------------------------------------")


'''grammar = r"""
    DN: {(<DET><N>|<DET>?<PN>)}     # DN -> (DET N| DET? PN) Defined Noun
    IN: {<N>}                       # UN -> N Undefined Noun
    NP:{<DN>}
    """ '''
grammar = r"""
    DN: {(<DET><N>|<DET><PN>)}     # DN -> (DET N| DET? PN) Defined Noun
    IN: {<N>}                       # UN -> N Undefined Noun
    NP:{<DN><DN>}
    """

tmp1, concated1=parsing(grammar, 'wordtagsall_sids.txt')
```

```
cases= caseFilter(tmp1, [2,6], ['nom gen', 'acc gen','acc acc', 'gen gen', 'nom nom'])
tmp= tmp1
concated= concated1
tmp=mergeLists(tmp, cases)
concated=mergeLists(concated, cases)

list2file(tmp, 'quran/All_tagged_terms_test2.txt', 'w')
list2file(concated, 'quran/All_tagged_terms_cooncated_test2.txt', 'w')
```

References

[1] E. S. Atwell *et al.*, “Understanding the Quran: A new grand challenge for computer science and artificial intelligence,” in *Proceedings of the GCCR’2010 Grand Challenges in Computing Research*, 2010.

[2] W. Wong, W. Liu, and M. Bennamoun, “Ontology Learning from Text: A Look Back and into the Future,” *ACM Computing Surveys*, vol. 44, no. 4, pp. 1–36, Aug. 2012.

[3] T. R. Gruber, “A Translation Approach to Portable Ontology Specifications,” *Appeared in Knowledge Acquisition*, vol. 5, no. 2, pp. 199–220, 1993.

[4] D. Sánchez, J. Cavero, and E. Martínez, “The road toward ontologies,” in *Integrated Series in Information Systems*, vol. 14, Springer, 2007, pp. 3–20.

[5] L. Drumond and R. Girardi, “A survey of ontology learning procedures,” *CEUR Workshop Proceedings*, vol. 427, 2008.

[6] P. Cimiano, *Ontology learning from text.* Springer US, 2006.

[7] F. Harrag, A. Al-Nasser, A. Al-Musnad, R. Al-Shaya, and S. Al-Salman, “Using association rules for ontology extraction from a Quran corpus,” in *Proc. 5th Int. Conf. Arabic Language Process.*, 2014, pp. 1–8.

[8] L. M. Bin Saleh and H. S. Al-Khalifa, “AraTation: An Arabic Semantic Annotation Tool,” in *Proceedings of the 11th International Conference on Information Integration and Web-based Applications*, 2009, pp. 447–451.

[9] M. G. H. Al Zamil and Q. Al-Radaideh, “Automatic extraction of ontological relations from Arabic text,” *Journal of King Saud University - Computer and Information Sciences*, vol. 26, no. 4, pp. 462–472, 2014.

[10] A. Benabdallah, · Mohammed, A. Abderrahim, and E. Abderrahim, “Extraction of terms and semantic relationships from Arabic texts for automatic construction of an ontology,” *International Journal of Speech Technology*, vol. 20, no. 2, pp. 289–296, 2017.

[11] N. F. Noy and D. L. McGuinness, “Ontology Development 101: A Guide to Creating Your First Ontology,” *Stanford Knowledge Systems Laboratory.* Stanford knowledge systems laboratory technical report KSL-01-05 and Stanford medical informatics technical report SMI-2001-0880, p. 25, 2001.

[12] M. Alrabiah, N. Alhelewh, A. Al-Salman, and E. Atwell, “An

Empirical Study On The Holy Quran Based On A Large Classical Arabic Corpus," *International Journal of Computational Linguistics (IJCL)*, vol. 5, no. 1, pp. 1–13, 2014.

[13] M. Al-Yahya, H. Al-Khalifa, A. Bahanshal, I. Al-Odah, and N. Al-Helwah, "An ontological model for representing computational lexicons a componential based approach," in *Proceedings of the 6th International Conference on Natural Language Processing and Knowledge Engineering, NLP-KE 2010*, 2010, pp. 1–6.

[14] A. D. Kashgary, "The paradox of translating the untranslatable: Equivalence vs. non-equivalence in translating from Arabic into English," *Journal of King Saud University - Languages and Translation*, vol. 23, no. 1, pp. 47–57, 2011.

[15] S. Saad, N. Salim, H. Zainal, and S. A. M. Noah, "A framework for Islamic knowledge via ontology representation," *Proceedings - 2010 International Conference on Information Retrieval and Knowledge Management: Exploring the Invisible World, CAMP'10*, pp. 310–314, Mar. 2010.

[16] A. Azman Ta'a, Syuhada Zainal Abidin, Mohd Syazwan Abdullah and and M. A. Bashah B Mat Ali, "Al-Quran Themes Classification Using Ontology," *Icoci.Cms.Net.My*, no. 74, pp. 383–389, 2013.

[17] H. Ullah Khan, S. Muhammad Saqlain, M. Shoaib, and M. Sher, "Ontology Based Semantic Search in Holy Quran," *International Journal of Future Computer and Communication*, vol. 2, no. 6, pp. 570–575, 2013.

[18] A. R. Yauri, R. A. Kadir, A. Azman, and M. A. A. Murad, "Ontology semantic approach to extraction of knowledge from Holy Quran," *2013 5th International Conference on Computer Science and Information Technology, CSIT 2013 - Proceedings*, pp. 19–23, 2013.

[19] Z. Yahya, M. T. Abdullah, A. Azman, and R. A. Kadir, "Query translation using concepts similarity based on Quran ontology for cross-language information retrieval," *Journal of Computer Science*, vol. 9, no. 7, pp. 889–897, Jul. 2013.

[20] M. A. Yunus, R. Zainuddin, and N. Abdullah, "Semantic query for Quran documents results," in *ICOS 2010 - 2010 IEEE Conference on Open Systems*, 2010, no. Icos, pp. 1–5.

[21] K. Dukes, "Statistical Parsing by Machine Learning from a Classical Arabic Treebank," PhD Thesis, School of Computing, University of Leeds, 2013.

[22] A. Muhammad, "Annotation of conceptual co-reference and text Mining the Qur'an," 2012.

[23] Z. Sardar, *What do Muslims believe?* Granta Books, 2011.

[24] F. Esack, *The Qur'an: A Short Introduction*. Oxford, United Kingdom: Oneworld; 12th edition edition, 2001.

[25] W. Wong, W. Liu, and M. Bennamoun, "Ontology Learning from Text: A Look Back and into the Future," *ACM Comput. Surv. Article*, vol. 44, no. 20, pp. 1–36, 2012.

[26] Y.-B. Kang, P. D. Haghighi, and F. Burstein, "CFinder: An intelligent key concept finder from text for ontology development," *Expert Systems with Applications*, vol. 41, no. 9, pp. 4494–4504, 2014.

[27] K. Sarkar, "A Hybrid Approach to Extract Keyphrases from Medical Documents," *International Journal of Computer Applications*, vol. 63, no. 18, pp. 975–8887, 2013.

[28] P. Buitelaar, P. Cimiano, and B. Magnini, "Ontology Learning from Text : An Overview," in *Learning*, IOS press, 2004, pp. 1–10.

[29] A. Maedche and S. Staab, "Learning Ontologies for the Semantic Web," *IEEE Intelligent systems*, vol. 16, no. 2, pp. 72–79, 2001.

[30] M. Hazman, S. R. El-Beltagy, and A. Rafea, "A Survey of Ontology Learning Approaches," *International Journal of Computer Applications*, vol. 22, no. 9, pp. 36–43, 2011.

[31] A. Gómez-pérez and D. Manzano-macho, "A survey of ontology learning methods and techniques OntoWeb Consortium," *Changes*, p. 86, 2003.

[32] M. Sabou, C. Wroe, C. Goble, and G. Mishne, "Learning Domain Ontologies for Web Service Descriptions: an Experiment in Bioinformatics," *Proceedings of the 14th international conference on World Wide Web*, no. Section 4, pp. 190–198, 2005.

[33] C. Y. Yong, R. Sudirman, K. M. Chew, and N. Salim, "Comparison of ontology learning techniques for Qur'anic text," *Proceedings - 2011 International Conference on Future Computer Sciences and Application, ICFCSA 2011*, pp. 192–196, Jun. 2011.

[34] B. Daille, "Variations and application-oriented terminology engineering," *International Journal of Theoretical and Applied Issues in Specialized Communication*, vol. 11, no. 1, pp. 181–197, 2005.

[35] T. Wandmacher, E. Ovchinnikova, U. Krumnack, and H. Dittmann, "Extraction, Evaluation and Integration of Lexical-Semantic Relations for the Automated Construction of a Lexical Ontology," in *Proceedings of the Third Australasian Workshop on Advances in Ontologies*, 2007, pp. 61–69.

[36] W. Wong, W. Liu, and M. Bennamoun, "Determination of Unithood and Termhood for Term Recognition," in *Handbook of Research on*

Text and web Technologies, 2009, p. 30.

[37] S. Boulaknadel, B. Daille, and D. Aboutajdine, “A multi-word term extraction program for Arabic language,” in *Language Resources and Evaluation Conference (LREC)*, 2008, pp. 3–6.

[38] H. Nakagawa and T. Mori, “A Simple but Powerful Automatic Term Extraction Method,” in *COLING-02 on COMPUTERM 2002: second international workshop on computational terminology*, 2002, pp. 1–7.

[39] P.-M. Ryu and K.-S. Choi, “An Information-Theoretic Approach to Taxonomy Extraction for Ontology Learning,” in *Ontology Learning from Text: Methods, Evaluation and Applications*, 2005, p. 15.

[40] S. R. El-Beltagy, A. Rafea, and I. D. Melamed, “KP-Miner: A keyphrase extraction system for English and Arabic documents,” *Information Systems*, vol. 34, no. 1, pp. 132–144, 2009.

[41] A. Sharaf and E. Atwell, “QurAna: Corpus of the Quran annotated with Pronominal Anaphora.,” in *LREC Language Resources and Evaluation Conference*, 2012, pp. 130–137.

[42] T. D. Wilson, “Models in information behaviour research,” *Journal of Documentation*, vol. 55, no. 3, p. 263, 1999.

[43] I. Bounhas and Y. Slimani, “A hybrid Approach for Arabic Multi - Word Term Extraction,” in *The IEEE International Conference on Natural Language Processing and Knowledge Engineering (IEEE NLP-KE)*, 2009, pp. 429–436.

[44] M. Attia, A. Toral, L. Tounsi, P. Pecina, and J. Van Genabith, “Automatic Extraction of Arabic Multiword Expressions,” in *The 7th Conference on Language Resources and Evaluation (LREC)*, 2010, pp. 19–27.

[45] F. Manola and E. Miller, “W3C Recommendation,” 2004. [Online]. Available: http: //www.w3.org/TR/rdf-primer/.

[46] P. Cimiano, C. Unger, and J. McCrae, “Ontology-Based Interpretation of Natural Language,” *Synthesis Lectures on Human Language Technologies*, vol. 7, no. 2, pp. 1–178, 2014.

[47] G. Lapalme, “XML: Looking at the Forest Instead of the Trees,” Montréal, Québec, Canada, 2013.

[48] N. Abbas and E. Atwell, “Qurany,” 2009. [Online]. Available: http://quranytopics.appspot.com/.

[49] G. G. Angeli, “LEARNING OPEN DOMAIN KNOWLEDGE FROM TEXT,” Stanford University, 2016.

[50] S. M. Alrehaili and E. Atwell, “Computational ontologies for semantic tagging of the Quran: A survey of past approaches,” in

Proceedings of the 2nd Workshop on Language Resources and Evaluation for Religious Texts, 2014, pp. 19–23.

[51] N. H. Abbas, “Quran’search for a Concept’Tool and Website,” Unpublished MSc Dissertation, School of Computing, University of Leeds, 2009.

[52] K. Dukes, “The Quranic Arabic Corpus,” *School of Computing, University of Leeds, UK*, 2011. [Online]. Available: https://scholar.google.co.uk/scholar?hl=en&q=quranic+arabic+corpus&btnG=&as_sdt=1,5&as_sdtp=#3.

[53] M. A. Sherif and A.-C. Ngonga Ngomo, “Semantic Quran A Multilingual Resource for Natural-Language Processing,” *Undefined*, vol. 1, pp. 1–5, 2009.

[54] M. Habash, *Mushaf Al Tajweed*. Dar-Al-Maarifah, Syria, 2001.

[55] “iszlam.net,” 2007. [Online]. Available: http://koran.iszlam.net/.

[56] H. S. Al-Khalifa, M. M. Al-Yahya, A. Bahanshal, and I. Al-Odah, “SemQ: A proposed framework for representing semantic opposition in the Holy Quran using semantic web technologies,” *Proceedings of the 2009 International Conference on the Current Trends in Information Technology, CTIT 2009*, pp. 44–47, 2009.

[57] A. R. Yauri, R. A. Kadir, A. Azman, and M. A. Azmi Murad, “Quranic-based concepts: Verse relations extraction using Manchester OWL syntax,” *Proceedings - 2012 International Conference on Information Retrieval and Knowledge Management, CAMP’12*, pp. 317–321, Mar. 2012.

[58] A. Sharaf and E. Atwell, “QurSim: A corpus for evaluation of relatedness in short texts.,” in *Lrec*, 2012, pp. 2295–2302.

[59] S. Saad, N. Salim, and H. Zainal, “Rules and Natural Language Pattern in Extracting Quranic Knowledge,” in *In the proceedings of the Taibah University International Conference on Advances in Information Technology for the Holy Quran and Its Sciences (NOORIC 2013)*, 2013, pp. 2–7.

[60] M. Al-Yahya and H. Al-Khalifa, “An Ontological Model for Representing Semantic Lexicons: An Application on Time Nouns in the Holy Quran,” *The Arabian Journal for Science and Engineering*, vol. 35, no. 2, pp. 21–35, 2010.

[61] T. Mukhtar, H. Afzal, and A. Majeed, “Vocabulary of Quranic concepts: A semi-automatically created terminology of Holy Quran,” in *2012 15th International Multitopic Conference, INMIC 2012*, 2012, pp. 43–46.

[62] K. T. Frantzi and S. Ananiadou, “The C-value/NC-value domain-independent method for multi-word term extraction,” *Journal of*

Natural Language Processing, vol. 6, no. 3, pp. 145–179, 1999.

[63] A. Hakkoum and S. Raghay, “Ontological approach for semantic modeling and querying the Qur’an,” 2015.

[64] A.-C. Ngonga Ngomo, “On Link Discovery using a Hybrid Approach,” *Journal on Data Semantics*, vol. 1, no. 4, pp. 203–217, 2012.

[65] S. Saad, N. Salim, and H. Zainal, “Pattern extraction for islamic concept,” *Proceedings of the 2009 International Conference on Electrical Engineering and Informatics, ICEEI 2009*, vol. 2, no. August, pp. 333–337, 2009.

[66] S. Saad and N. Salim, “Methodology of Ontology Extraction for Islamic Knowledge Text,” in *Postgraduate Annual Research Seminar*, 2008.

[67] S. Saad, N. Salim, and S. Zainuddin, “An early stage of knowledge acquisition based on Quranic text,” *2011 International Conference on Semantic Technology and Information Retrieval, STAIR 2011*, no. June, pp. 130–136, 2011.

[68] X. Jiang and A. H. Tan, “CRCTOL: A semantic-based domain ontology learning system,” *Journal of the American Society for Information Science and Technology*, vol. 61, no. 1, pp. 150–168, 2010.

[69] N. I. Al-Rajebah, H. S. Al-Khalifa, N. I. Al-Rajebah, and H. S. Al-Khalifa, “Extracting Ontologies from Arabic Wikipedia: A Linguistic Approach,” *Arabian Journal for Science and Engineering*, vol. 39, pp. 2749–2771, 2014.

[70] M. Hadni, A. Lachkar, and S. A. Ouatik, “MULTI-WORD TERM EXTRACTION BASED ON NEW HYBRID APPROACH FOR ARABIC LANGUAGE,” in *the Second International Conference on Computational Science and Engineering (CSE-2014)*, 2014, pp. 109–120.

[71] A. Al-Arfaj and A. Al-Salman, “Towards Concept Extraction for Ontologies on Arabic language,” *International Journal on Islamic Applications in Computer Science And Technology (IJASAT)*, vol. 4, no. 4, pp. 9–19, 2016.

[72] F. Xu, D. Kurz¼, J. Piskorski, and S. Schmeier¼, “An Domain Adaptive Approach to Automatic Acquisition of Domain Relevant Terms and their Relations with Bootstrapping,” in *the Third International Conference on Language Resources an Evaluation (LREC’02)*, 2002.

[73] A. Maedche and S. Staab, “Mining Ontologies from Text Alexander,” in *12th International Conference, EKAW2000*, 2000, pp. 189–201.

[74] M. Missikoff, R. Navigli, and P. Velardi, “The Usable Ontology: An Environment for Building and Assessing a Domain Ontology,” in *The Semantic Web - ISWC 2002*, I. Horrocks and J. Hendler, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 39–53.

[75] P. Cimiano and J. Völker, “Text2Onto,” in *Natural Language Processing and Information Systems*, E. Montoyo, Andrés and Muńoz, Rafael and Métais, Ed. Springer Berlin Heidelberg, 2005, pp. 227–238.

[76] S. Zaidi, A. Abdelali, F. Sadat, and M.-T. Laskri, “Hybrid Approach for Extracting Collocations from Arabic Quran Texts,” 2012.

[77] M. Alhawarat, “Extracting Topics from the Holy Quran Using Generative Models,” *INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS*, vol. 6, no. 12, pp. 288–294, 2015.

[78] Al-mishkat, “al-mishkat,” 2012. [Online]. Available: http://www.al-mishkat.com/words/.

[79] S. R. El-Beltagy and A. Rafea, “KP-Miner: Participation in SemEval-2,” in *Proceedings of the 5th international workshop on semantic evaluation*, 2010, pp. 190–193.

[80] S. Saad, N. Salim, and N. Omar, “Keyphrase extraction for islamic knowledge ontology,” in *Proceedings - International Symposium on Information Technology 2008, ITSim*, 2008.

[81] S. F. Ali, Abobaker and Brakhw, M Alsaleh and Nordin, Munif Zarirruddin Fikri Bin and ShaikIsmail, “Some Linguistic Difficulties in Translating the Holy Quran from Arabic into English,” *International Journal of Social Science and Humanity*, vol. 2, no. 6, pp. 588–590, 2012.

[82] K. Dukes and E. Atwell, “LAMP : A Multimodal Web Platform for Collaborative Linguistic Analysis,” *Lrec 2012*, pp. 3268–3275, 2012.

[83] Y. Marton, N. Habash, and O. Rambow, “Dependency Parsing of Modern Standard Arabic with Lexical and Inflectional Features,” *Computational Linguistics*, vol. 39, no. 1, pp. 161–194, 2013.

[84] M. Collins, J. Haj, L. Ramshaw, and C. Tillmann, “A Statistical Parser for Czech*,” in *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics*, 1999, pp. 505–512.

[85] Y. Marton, N. Habash, and O. Rambow, “Improving Arabic Dependency Parsing with Lexical and Inflectional Morphological Features,” in *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, 2010, pp. 13–21.

[86] X. Jiang and A. H. Tan, “CRCTOL: A semantic-based domain ontology learning system,” *Journal of the American Society for Information Science and Technology*, 2010.

[87] S. Staab and R. Studer, *Handbook on Ontologies*. Springer, 2007.

[88] G. A. Miller, “WordNet: A Lexical Database for English,” *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[89] F. M. Suchanek, G. Kasneci, and G. Weikum, “YAGO: A Large Ontology from Wikipedia and WordNet,” *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 6, no. 3, pp. 203–217, 2008.

[90] C. Fellbaum, *An Electronic Lexical Database*. Wiley Online Library, 1998.

[91] G. Bordea, E. Lefever, and P. Buitelaar, “SemEval-2016 Task 13: Taxonomy Extraction Evaluation (TExEval-2),” in *SemEval*, 2016, pp. 1081–1091.

[92] B. Zafar, M. Cochez, and U. Qamar, “Using Distributional Semantics for Automatic Taxonomy Induction,” in *2016 International Conference on Frontiers of Information Technology (FIT)*, 2016, pp. 348–353.

[93] A. Hippisley, D. Cheng, and K. Ahmad, “The head-modifier principle and multilingual term extraction,” *Natural Language Engineering*, vol. 11, no. 2, pp. 129–157, 2017.

[94] M. A. Hearst, “Automatic acquisition of hyponyms from large text corpora,” in *Proceedings of the 14th conference on Computational linguistics-Volume 2*, 1992, pp. 539–545.

[95] P. S. Jacobs, G. R. Krupka, and L. F. Rau, “LEXICO-SEMANTIC PATTERN MATCHING AS A COMPANION TO PARSING IN TEXT UNDERSTANDING,” in *Workshop on Speech and Natural Language colocated with the 6th Human Language Technology Conference*, 1991, pp. 337–341.

[96] W. IJntema, J. Sangers, F. Hogenboom, and F. Frasincar, “A lexico-semantic pattern language for learning ontology instances from text,” *Journal of Web Semantics*, 2012.

[97] R. Navigli, P. Velardi, S. Faralli, D. Informatica, and S. Universit, “A Graph-based Algorithm for Inducing Lexical Taxonomies from Scratch,” *… of the Twenty-Second international joint …*, 2009.

[98] N. Ghneim, M. Al, and S. Ali, “Building a Framework for Arabic Ontology Learning,” in *Knowledge Man- agement and Innovation in Advancing Economicss: Analyses & Solutions*, 2009, p. 1730–1 1735.

[99] L. Al-Safadi, M. Al-Badrani, and M. Al-Junidey, “Developing

Ontology for Arabic Blogs Retrieval," *International Journal of Computer Applications*, vol. 19, no. 4, pp. 975–8887, 2011.

[100] N. Y. Habash and G. Hirst, "Introduction to Arabic Natural Language Processing Introduction to Arabic Natural Language Processing Synthesis Lectures on Human Language Technologies Editor Introduction to Arabic Natural Language Processing," *Synthesis Lectures on Human Language Technologies*, vol. 3, pp. 1–187, 2010.

[101] M. A. Attia, "Handling Arabic Morphological and Syntactic Ambiguity within the LFG Framework with a View to Machine Translation," 2008.

[102] K. Dukes, A. Sharaf, and E. Atwell, "Online visualization of traditional quranic grammar using dependency graphs," *The Foundations of …*, pp. 1–15, 2010.

[103] M. Eid, "Verbless Sentences in Arabic and Hebrew," in *Perspectives in Arabic Linguistics III*, C. Bernard and M. Eid, Eds. Amsterdam: Benjamins, 1991, pp. 31–61.

[104] M. Eid, "THE COPULA FUNCTION OF PRONOUNS*," *Lingua*, vol. 59, pp. 197–207, 1983.

[105] E. Frick, C. Schnober, and P. Bá, "Evaluating Query Languages for a Corpus Processing System," in *The eighth international conference on Language Resources and Evaluation (LREC)*, 2012, pp. 2286–2294.

[106] T. Buckwalter, "Buckwalter Arabic Morphological Analyzer Version 2.0," *LDC2004L02*, 2004. .

[107] L. Aldhubayi, "Unified Quranic Annotations and Ontologies," University of Leeds, 2012.

[108] S. M. Alrehaili and E. Atwell, "Discovering Qur'anic Knowledge through AQD: Arabic Qur'anic Database, a Multiple Resources Annotation-level Search," in *2nd IEEE International Workshop on Arabic & derived Script Analysis and Recognition (ASAR)*, 2018, pp. 102–107.

[109] K. Dukes and N. Habash, "Morphological Annotation of Quranic Arabic.," in *Lrec*, 2010, pp. 1–3.

[110] B. Sadeghi, "The Chronology of the Qurʾān: A Stylometric Research Program," *Arabica*, vol. 58, pp. 210–299, 2011.

[111] S. M. Alrehaili, "The Chronology of the Texts in the Holy Quran According to NLP," Unpublished MSc Dissertation, School of Computing, University of Leeds, 2012.

[112] G. Stragand, "Simple Fuzzy String Similarity in Java," 2011. .

[113] J. Paul, “Nouvelles Recherches Sur La Distribution Florale,” *Bulletin de la Société vaudoise des Sciences Naturelles*, vol. 44, pp. 223–270, 1908.

[114] D. Janus and A. Przepiórkowski, “Poliqarp An open source corpus indexer and search engine with syntactic extensions,” in *the 45th Annual Meeting of the Association for Computational Linguistics*, 2007, pp. 85–88.

[115] T. Arts, Y. Belinkov, N. Habash, A. Kilgarriff, and V. Suchomel, “arTenTen: Arabic Corpus and Word Sketches,” *Journal of King Saud University - Computer and Information Sciences*, vol. 26, pp. 357–371, 2014.

[116] J. Sager, D. Dungworth, and P. Mcdonald, *English Special Languages: Principles and Practice in Science and Technology*. John Benjamins Pub Co, 1980.

[117] S. M. Alrehaili and E. Atwell, “Extraction of Multi-Word Terms and Complex Terms from the Classical Arabic Text of the Quran,” *International Journal on Islamic Applications in Computer Science And Technology*, vol. 5, no. 3, pp. 15–27, 2017.

[118] X. Jiang and A.-H. Tan, “CRCTOL: A semantic-based domain ontology learning system,” *Journal of the American Society for Information Science and Technology*, vol. 61, no. 1, pp. 150–168, Jan. 2010.

[119] Ibn-Katheer, *Tafseer Al-Qur’an*, 1st ed. Dar Taibah, 2008.

[120] K. Dukes, E. Atwell, A. -Baquee, and M. Sharaf, “Syntactic Annotation Guidelines for the Quranic Arabic Dependency Treebank,” in *LREC’2010 Language Resources and Evaluation Conference.*, 2010.

[121] W. Wong, W. Liu, and M. Bennamoun, “Determining the Unithood of Word Sequences using Mutual Information and Independence Measure,” 2008.

[122] A. A. Ti and M. Zhang, “Term Extraction Through Unithood And Termhood Unification,” in *Int’l joint conf on Natural Language Proc*, 2008.

[123] J. Perkins, *Python text processing with NLTK 2.0 cookbook*. Birmingham, UK: Packt Publishing Ltd, 2010.

[124] N. Hardeniya, *NLTK Essentials Build cool NLP and machine learning applications using NLTK and other Python libraries*. Packt Publishing Ltd, 2015.

[125] S. Alrehaili and E. Atwell, “A hybrid-based Term Extraction method using the Arabic text of the Qur’an,” in *4th International Conference on Islamic Applications in Computer Science and*

Technologies (IMAN 2016), 2016, pp. 20–22.

[126] F. Alotaiby, S. Foda, and I. Alkharashi, “Clitics in Arabic Language: A Statistical Study *,” in *24th Pacific Asia Conference on Language, Information and Computation*, 2010, pp. 595–601.

[127] M. Hall Eibe Frank, G. Holmes, B. Pfahringer Peter Reutemann, and I. H. Witten, “The WEKA Data Mining Software: An Update,” *SIGKDD Explor. Newsl.*, vol. 11, no. 1, pp. 10–18, 2009.

[128] A. Jones, *Arabic through the Qur’an*. Islamic Texts Society, 2005.

[129] E. Lefever, “LT3: A Multi-modular Approach to Automatic Taxonomy Construction,” in *9th International Workshop on Semantic Evaluation (SemEval 2015)*, 2015, pp. 944–948.

[130] M. Horridge and S. Bechhofer, “The OWL API: A Java API for OWL Ontologies,” *Semantic Web journal*, vol. 2, no. 1, pp. 11–21, 2011.

[131] S. M. Alrehaili, M. Alqahtani, and E. Atwell, “A Hybrid Methods of Aligning Arabic Qur’anic Semantic Resources,” in *2nd IEEE International Workshop on Arabic & derived Script Analysis and Recognition (ASAR)*, 2018, pp. 108–113.