Answer Selection for Questions with Multiple-Answer-Choices in Arabic Question Answering System Based on Textual Entailment Recognition

Sélection de réponses pour les questions à choix multiples en arabe; système de réponses aux questions fondées sur la reconnaissance de l'Implication textuelle

A Thesis Submitted to the Department of Mathematics and Computer Science of the Royal Military College of Canada

By

Anes Enakoa

In Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy

April, 2019

This thesis is dedicated to my mother and my family for their love, care and support.

Acknowledgments

First and above all, I would like to thank the **Almighty God** for giving me the knowledge, ability and opportunity to undertake this research study. Of course, this outcome would not have been possible without the help and support from Him.

I am most grateful to my supervisor **Dr.Yawei Liang**, for supervising this thesis, keeping motivating me and for having provided me with tireless guidance and moral support during all these years. I would like to express my gratitude to my family particularly my mother and my wife, professors, committees, students, colleagues, and friends. My thanks are also due to the ministry of higher education and scientific research of Libya, for their financial support to study in Canada.

Abstract

Question Answering (QA) system is one of the most important and demanding tasks in the field of Natural Language Processing (NLP), which is concerned with answering questions posed in a natural language. In QA systems, the answer generation task generates a list of candidate answers to the user's question, in which only one answer is correct. Answer Selection is one of the main components of the QA, which is responsible for selecting the best answer choice from the candidate answers suggested by the system. However, the selection process can be very challenging especially in Arabic due its particularities. To address this challenge, we propose an approach to answer questions with multiple answer choices for Arabic QA systems based on Textual Entailment (TE) recognition. The developed approach employs Support Vector Machine (SVM) classifier that considers lexical, semantic and syntactic features in order to recognize the entailment between the posed question and the candidate answers. A set of experiments has been conducted to measure the effectiveness of our method. The obtained results show that our method helps significantly to tackle the problem of Answer Selection in Arabic Question Answering system.

Résumé

Le système de réponse aux questions (RQ) est l’une des tâches les plus importantes et les plus exigeantes dans le domaine du traitement du langage naturel (TLN). Il fait référence à la réponse à des questions posées dans un langage naturel. Dans les systèmes d'assurance qualité, la tâche de génération de réponses génère une liste de réponses de candidats à la question de l'utilisateur, dans laquelle une seule réponse est correcte. La sélection des réponses est l’un des composants principaux de l’RQ, qui est responsable de la sélection du meilleur choix de réponse parmi les réponses suggérées par le système. Cependant, le processus de sélection peut être très difficile, en particulier en arabe, en raison de ses particularités. Pour relever ce défi, nous proposons une approche permettant de répondre à des questions à choix multiples pour les systèmes d'assurance qualité en arabe qui sont fondés sur la reconnaissance de l’implication textuelle (IT). L'approche combine trois ensembles de fonctionnalités, à savoir le lexique, la sémantique et la syntaxe. Elle évalue si l'une des réponses candidats peut être déduite du texte renvoyé par le système. Une série d'expériences a été menée pour mesurer l'efficacité de notre méthode. Les résultats obtenus démontrent que notre méthode aide de manière significative à résoudre le problème de la sélection des réponses dans le système de réponses aux questions en arabe.

Table of Contents

List of Tables

List of Figures

List of Abbreviations and Acronyms

Abbr	Full form
ArbTEDS	Arabic TE dataset
AS	Answer Selection
AWN	Arabic WordNet
BM	Bigram Match
BP	Broken Plurals
CoNLL	Conference on Natural Language Learning
H	Hypothesis
IDRAAQ	Information and Data Reasoning for Answering Arabic Questions
IR	Information Retrieval
ISRI	Information Science Research Institute
CLEF	Cross Language Evaluation Forum
JIRS	Java Information Retrieval System
LCH	Leacock & Chodorow
LCS	Longest Common Subsequence
MADA	Morphological Analysis And Disambiguation for Arabic
ML	Machine Learning
MRR	Mean Reciprocal Rank
MSA	Modern Standard Arabic
MSTParser	Minimum Spanning Tree Parser
NE	Named-Entity
NEM	Named Entity Matching
NER	Named Entity Recognition

NLP	Natural Language Processing
NP	Noun Phrase
OBJ	Object-Verb
PoS	Part-of-Speech
PR	Passage Retrieval
PWN	Princeton WordNet
QCRI	Qatar Computing Research Institute
QE	Query Expansion
QA	Question Answering
QA4MRE	Question Answering for Machine Reading Evaluation
SE	Search Engine
SP	Sound Plural
SBJ	Subject-verb
SUMO	Suggested Upper Merged Ontology
SVM	Support Vector Machine
T	Text
TREC	Text REtrieval Conference
TE	Textual Entailment
TM	Trigram Match
UM	Unigram Match
WWW	World Wide Web
WUP	Wu and Palmer
XML	Extensible Markup Language

Arabic Transliterations[1]

Letter	HSB	BW	Unicode name	Letter	HSB	BW	Unicode name
ء	’	’	Hamza	ظ	Ď	Z	Za’
آ	Ā	|	Alif-Madda above	ع	ς	E	Ayn
أ	Â	>/O	Alif-Hamza above	غ	γ	g	Ghayn
ؤ	ŵ	&/W	Waw-Hamza above	ـ	_	_	Tatweel
إ	Ǎ	</I	Alif-Hamza below	ف	f	f	Fa’
ئ	ŷ	}	Ya’-Hamza above	ق	q	q	Qaf
ا	A	A	Alif	ك	k	k	Kaf
ب	b	b	Ba’	ل	l	l	Lam
ة	ħ	p	Ta’-Marbuta	م	m	m	Meem
ت	t	t	Ta’	ن	n	n	Nun
ث	θ	v	Tha’	ه	h	h	Ha’
ج	j	j	Jeem	و	w	w	Waw
ح	H	H	Ha’	ى	ý	Y	Alif-Maqsura
خ	x	x	Kha’	ي	y	y	Ya’
د	d	d	Dal	Arabic diacritics			
ذ	ð	*	Dhal	ـَ	a	a	Fatha, i.e. /a/
ر	r	r	Ra’	ـُ	u	u	Damma, i.e. /u/
ز	z	z	Zay	ـِ	i	i	Kasra, i.e. /i/
س	s	s	Sen	ـً	ã	F	Fathatan, i.e. /an/
ش	š	$	Shen	ـٌ	ũ	N	Dammatan, i.e. /un/
ص	S	S	Sad	ـٍ	ĩ	K	Kasratan, i.e. /in/
ض	D	D	Dhad	ـّ	~	~	Shadda
ط	T	T	Ta’	ـْ	.	o	Sukun (zero vowel)

[1] The translation used in this document is Buckwalter (BW) Arabic transliteration [33]

Chapter 1

Introduction

In the last few decades, the amount of information available on the Internet has been increasing remarkably. The Web has become the main source of all kind of data stored in electronic format. Getting precise information in real time is becoming increasingly difficult [57]. Current Information Retrieval (IR) systems and search engines such as Google[1] and Yahoo[2] do not allow users to return concise answers to their questions. Given some keywords, the system only returns the relevant ranked documents that contain these keywords and the user has to take the trouble of searching for the answers inside each document. In many cases, this method consumes the time of the users and does not help them to get the direct relevant information efficiently from very big group of documents. In fact, users often have specific questions in their mind. They would like to express their questions in their natural language without being restricted to a particular query language and want precise answers to those questions [22].

Question Answering (QA) system addresses this problem. The main goal of QA is to provide inexperienced users with answers to questions

[1] https://www.google.ca/

[2] https://www.yahoo.com/

rather than full documents. While the classical information retrieval systems present the users with a set of documents that relate to user questions without indicating the exact correct answers, the QA system enables the user to ask questions immediately in their native language and get precise and direct answers, which saves a lot of time and effort for the user. The QA systems are fed with the questions in natural language by the user as input, the systems search matching answers in set of documents and return the concise answers to the questions as output [106][29][57][73]. For instance, given the question “When was the Royal Military College established?” A QA system should instantly return the exact answer: 1876. Table 1.1 shows the main differences between the conventional IR and QA.

Due to the fact that the amount of Arabic content on the Internet has been extremely increasing and that regular IR techniques cannot satisfy the user's information need, the need to reliable Arabic QA systems is becoming crucial. However, the research and development in the area of Arabic QA can be considered as lagging behind compared to similar work on non-Arabic systems [103]. A lot of research has been done to build QA systems for English and other Latin-based languages. On the other hand, very little work has been made by researchers to reach an acceptable level in the Arabic QA task.

Table 1.1: The differences between IR and QA [18]

	IR	**QA**
Input	Keywords	Natural language question
Output	A list of documents	Phrases and Words having the answer

1.1 Answer Selection Task

In QA systems, the answer generation task generates a list of candidate answers to the user's question, in which only one answer is correct. Answer Selection is one of the main components of the QA, which is concerned with selecting the best answer choice from the candidate answers suggested by the system. The problem of answer selection in Question Answering system can be formulated as the following:

Given a question q, a set of candidate answers $\{a_1, a_2, ..., a_n\}$ and a supporting text t , the goal is to choose the correct answer a_i.

The selection process can be very challenging especially in Arabic due its particularities. Unlike languages such as English, Spanish, French or Italian, Arabic differs in its richness and complexity that needs special handling to make reliable QA systems. The challenges are not limited to those commonly faced by non-Arabic systems. Each integrated component in the Arabic QA system may have a negative impact on the performance of the system unless the particularities of this language are considered. For Latin languages, the task of answer selection and validation has been studied to a great extent and many approaches have been proposed to tackle the problem, but for Arabic, the work has been very limited and most of the research in the area of Arabic Question Answering tends to focus on the information retrieval step rather than the answer selection step, which makes them very similar to traditional Question Answering systems.

1.2 Textual Entailment

Textual Entailment (TE) is one of the important natural language processing challenges. Given two expressions, one is called the “Text” and denoted as T and the other one is called the “Hypothesis” and denoted as H, TE determines whether the meaning of the “Hypothesis” could be entailed by the meaning the “Text”. This means that a human would agree that the meaning of *T* implies the meaning of *H*. More formally, a text *T* entails a hypothesis *H*, if *H* is true in every circumstance, in which *T* is true [40]. For example, the text *T*= “Mark’s wife is beautiful” entails the hypothesis *H* = “Mark is married”. Likewise, *T*= “Adam has worked in Libya” entails *H* = “Adam has worked in an Arabic country”. On the other hand, *T*= “Adam has worked in an Arabic country” does not entail *H* = “Adam has worked in Libya”.

Recognizing the entailment between two texts is important in many natural language processing applications where a problem can be formulated in terms of TE, such as information retrieval, summarization, machine translation and question answering. Recently, number of studies has been addressing the problem of answer selection in QA using TE techniques in non-Arabic language. Our investigation and study of the advances in the field showed us that adopting TE techniques has had a significant improvement on the performance of the QA systems in English and other languages. The objective of this work is to study the suitability and the effectiveness of these techniques for improving the answer selection in Arabic QA systems.

To address the challenge of answer selection in Arabic QA systems, we propose an approach to answer questions with multiple answer choices for Arabic. The approach is based on Textual Entailment (TE) recognition method. The basic idea is to evaluate whether one of the candidate answers can be inferred from the text returned by the system. In case of a candidate answer is being entailed by the supporting text, it then can be chosen as a correct answer. The developed approach employs a Support Vector Machine (SVM) that considers lexical, semantic and syntactic features in order to recognize the entailment between the generated hypotheses (H) and the text (T). Each retrieved sentence is considered as text (T) and paired with the corresponding hypothesis (H) to represent T-H pair. Thereafter, features are extracted from the T-H pairs and fed into the classifier in order to classify new samples based on the trained model.

1.3 Thesis Objective

Arabic differs from Latin languages both syntactically and morphologically. The particularities of the Arabic and the high level of its complexity add extra challenges to NLP applications in general and to QA specifically. To achieve the task of answer selection in QA, different language processing resources and tools are required. However, most of the existing NLP tools are developed for Latin languages and not completely suitable for Arabic. Given that situation, the proposed work attempts to address the following research question: Is it possible to develop a new model based on TE recognition that combines lexical, semantic and syntactic features in one approach to solve the problem of Answer Selection in Arabic QA?

1.4 Contributions

The work described in this research has achieved several goals. The main contributions can be summarized as follows:

- Study the advances in the field of Question Answering systems, Textual Entailment recognition and investigate the suitability and the effectiveness of applying different techniques for improving the answer selection in Arabic QA systems.

- Achieving the research goal to build an Answer Selection model for QA system that performs better than the state-of-the-art Arabic QA systems.

- Introducing an approach to answer questions with multiple-answer-choices for Arabic QA systems based on Textual Entailment (TE) recognition.

- Our work is the first work in Arabic Question Answering that combines three different sets of features that include lexical, semantic and syntactic features in one approach to solve the problem of textual entailment recognition in Arabic.

- Utilizing different Arabic resources and tools and performing multiple kinds of preprocessing in order to tackle the Arabic challenges and to achieve our goals.

- Conducting a set of experiments with different types of questions using different datasets; analysing the obtained results and comparing our work to similar Arabic systems.

1.5 Organisation of Thesis

The rest of the dissertation is structured as follows: Chapter 2 presents a background and gives an overview of literature that relates to Arabic QA systems. Literature review is divided into three parts: The first part introduces the advances in the history of Arabic QA systems giving more attention to answer selection task. The second part talks about QA4MRE @ CLEF[3] campaign and describes the participated Arabic systems. The third part discusses the utilization of Textual Entailment approaches in Arabic QA systems. Chapter 3 discusses the general architecture of QA systems. The chapter describes the most common pipeline architecture that most of the QA systems share. Later in this chapter, we provide a brief description

[3] http://www.clef-campaign.org/

of different categories of QA system based on some criteria, the appropriate evaluation metrics that used by the QA community researchers to assess and compare their work as well as the standard QA evaluation forums that available to support evaluating different systems. Chapter 4 has been divided to two sections: The first section explains the importance of Arabic and how it differs from Indo-European languages. The significant challenges faced by researchers to build many natural language processing applications in general and QA specifically are discussed. The second section presents the main tools have been used in this research. Chapter 5 talks about using machine learning to solve the problem of recognizing the entailment between the text and the hypothesis. Thereafter, a detailed description about the selected features and the approach we applied to model the textual entailment as classification problem are provided. Chapter 6 presents the proposed approach to answer questions with multiple-answer-choices for Arabic QA systems based on Textual Entailment (TE) recognition. The core modules of the system are outlined and each module of these modules consists of number of submodules are also described in details. Chapter 7 provides a discussion about the experiments and the results of applying our approach of Answer Selection through Textual Entailment over Arabic texts. We started with an in-depth description of the datasets and the measures were used for the evaluation. Thereafter, the conducted experiments are presented and the results are reported and analysed. We end up with conclusions and future work in Chapter 8.

Chapter 2

Literature Review

A lot of research has been done to build QA systems for English and other Latin-based languages. On the other hand, very little work has been made by researchers to reach an acceptable level in the Arabic QA task. This is due to nature of Arabic language itself as well as the challenges faced by Arabic systems (which are explained in Chapter 4). The challenges are not limited to those ones commonly faced by non-Arabic systems. Each integrated component in the Arabic QA system may have a negative impact on the performance of the system unless the particularities of this language are considered. Recently, number of studies has been addressing the problem of answer selection in QA using TE techniques in non-Arabic language. However, the research in the area of Arabic QA has tended to focus on the information retrieval step rather than the answer selection step, which makes them lagging behind comparing with other non-Arabic QA systems.

This chapter presents a review that relates to Arabic QA systems. The review is divided into three parts: The first part introduces the advances in the history of Arabic QA systems giving more attention to answer selection task. The second part talks about QA4MRE @ CLEF[1] campaign and describes the participated Arabic systems. The third part discusses the utilization of Textual Entailment approaches in Arabic QA systems.

[1] http://www.clef-campaign.org/

2.1 Advances in Arabic Question Answering

Historically, one of the first attempts to tackle the problem of Arabic Question Answering was a system called AQAS. It was developed by Mohammed et al., in 1993 [90]. AQAS is a closed domain knowledge-based system that retrieves answers from only structured data and not from raw text written in natural language. It is fed by queries that follow pre-defined rules and matches them against frames in a knowledge base. The developers of AQAS have not presented their system's experimental results.

After almost a decade of advancement in the field of Arabic natural language processing and information retrieval, Hammo et al. [61] designed and implemented their QA system called QARAB. It was the first Arabic QA system that used sophisticated Natural Language Processing techniques such as POS tagging, NER, and lexicon based stemming to parse the user's query and the documents to identify the candidate passages for answer selection. QARAB accepts queries expressed in Arabic language and returns short passages that are likely to contain an answer to the question rather than retrieving the direct answer. The system's primary source of knowledge is a collection of Arabic newspaper texts "Al-Raya", a newspaper printed in Qatar. It is considered a closed system because it works based on three assumptions. First, the answer to the question of the user is contained in its collection. Second, the answer can only be found in one document in that collection. Third, the answer is a short passage. QARAB is based on a set of rules for each question type, but does not handle questions of types the *How* and *Why* since they require more advanced processing. QARAB uses shallow language understanding and treats questions as a bag of words and did not understand the content of the question at a deep level. Experiments have been conducted by four native speakers who checked the correctness of QARAB answers using 113 questions as a test-bed. The developers reported that the results showed recall and precision of 97.3%.

In 2007, Benajiba et al. [28] developed a QA system specifically for Arabic factoid questions. ArabiQA consists of three modules, Passage Retrieval, Named Entities Recognition and Answer Extraction. It also integrates JIRS (Java Information Retrieval System) to extract passages from Arabic texts. An evaluation corpus on the basis of CLEF[2] guidelines was prepared to test the system. The authors reported a precision of 83.3%, but the details were not given. In spite of that the system has been designed for an open domain, but it has not been tested in such an environment.

While most of the Arabic QA systems were built to handle factoid questions, in 2009, Brini et al. [32] made an attempt for building an Arabic QA system to deal with both factoid and definition questions. The system is named QASAL (Question Answering System for Arabic Language). It employs the NooJ[3] platform as a linguistic development environment and takes advantage of some linguistic techniques from IR and NLP to process Arabic text documents to extract the precise answers which requested by users. Google search engine was used as Web resource to answer 43 definition questions. According to the authors, the preliminary results obtained for the definition questions have a precision equal to 94% and recall equal to 100%.

Kanaanet et al. [68] described another QA system for short Arabic questions. To achieve its task, the system uses data redundancy rather than complicated linguistic analyses for questions and candidate answers. This system does not support *How* and *Why* questions because of the complex processing involved in handling such questions. The authors tested their system using a collection consisting of 25 documents from the Internet and

[2] http://www.clef-campaign.org/

[3] http://www.nooj4nlp.net

12 questions. They do not mention why and how these questions and documents have been selected. Authors have not compared their results to any previously developed Arabic QA systems.

ArQA is another QA system that was introduced in 2011 by Abdelbaki and Shaheen [1] to handle Arabic factoid questions expressed in natural language. The system gives more attention to the question analysis process by identifying the question focus for each question. Then the system uses the semantic similarity between the question focus and the candidate answer to recognize the answers [103][109]. The architecture of this system consists of four modules: Question Processing, Passage Retrieval, Answer Extraction and Answer Validation. Each module uses IR and NLP techniques and tools to enhance validity of retrieved answers [35].

Bekhti and Al-Harbi [27] also recognized the importance of the question analysis phase and its impact on the performance of the whole system. They proposed an Arabic QA system named AQuASys. The system consists of three modules: Question Analysis, Sentences Filtering and Answer Extraction. It deals with unformatted questions written in an Arabic natural language. The user's question words are classified into three classes: interrogative noun, verbs and question's keywords. NLP techniques were applied to analyze the question posed by the user in order to generate informative and valuable features from it. These features had a strong effect on answer finding accuracy performance. In order to assess the system's performance, the developers used a corpus from ANERCorp[4] and ANERgazet which are available online along with a set of 80 questions. The authors indicated that they obtained 66.25 % in precision and 97.5 % in recall.

QArabPro [14] is another Arabic QA system designed to deal with all types of queries including questions of type "*How*" and "*Why*". The system

[4] http://users.dsic.upv.es/~ybenajiba/downloads.html

is rule-based that uses a set of rules for each type of *WH* questions. The overall accuracy of the system was 84%, but for these two types of questions was low, 62% for *Why* questions and 69% for *How* questions.

Unlike other Arabic question answering systems, JAWEB is a web-based QA system developed by Kurdi et al. [76]. The system consists of four modules: user interface, question analyzer, passage retrieval and answer extractor. For answer extraction task, the system module uses scoring formulas to measure the similarity between the user's query and the retrieved sentences. The answers are ordered based on of their relevance to the given question and the answer that obtains the highest score is selected as the true answer. For evaluation, the authors compared their system to the web-based QA system ask.com[5] and they reported 15-20% higher recall with average of 100% recall and 80% precision. The system does not use a Named Entity Recognition to identify named entities. In addition, the *Why* and *How* questions are not handled.

AlQuAnS [95] is an Arabic QA System that includes an online and offline parts. The offline part consists of two modules: preprocessing and semantic interpreter modules. In the online part, the system contains four modules: preprocessing, question Analysis, information retrieval and answer extraction modules. The offline part has two components, the preprocessing and semantic interpreter modules. Each module of these modules composed of some submodules that are responsible for fulfilling other subtasks. For answer selection, the system uses answer patterns provided by the pattern construction module to extract the proper answer from the retrieved sentences. The patterns are built from the training dataset using a set of features. Both the online and the offline versions of the system were compared with the system presented by Abouenour et al. [5]. The online part achieved 26.15%, 12.57% and 45.97% in accuracy, MRR and

[5] https://www.ask.com/

answered questions respectively while the Offline system reached 22.20%, 8.16 %and 47.66% in accuracy, MRR and answered questions respectively.

2.2 Answer Selection and QA4MRE

Since 2011, The Conference and Labs of the Evaluation Forum (CLEF)[6] started Question Answering for Machine Reading Evaluation (QA4MRE). The goal behind introducing QA4MRE task is to give more attention to reading comprehension and make participating systems concentrate on answer selection and validation and skip the answer generation task.

The QA4MRE task focuses on taking a single document and a set of questions as input and returning an exact answer as output. Questions are in the form of multiple choices. Each question has 5 different options and only one answer is correct [100]. The detection of the correct answer is specifically designed to require various types of inference, and a deeper level of text understanding [116]. The task introduced to evaluate how the computer understands a comprehension passage in the same way that reading comprehension tests designed to evaluate how well a human can understand a text. By providing a single evaluation platform for the experimentation, the QA4MRE encourages the interest in this research line and pays more attention to the task of answer selection and validation over the information retrieval based tasks in QA [64].

The main task of the competition consisted of four topics: Music and Society, Climate Change, AIDS and Alzheimer's (sources: blogs, web, news). Each topic had four reading tests. Each reading test provided with one single document followed by 10 questions and a set of five choices per question. The total set included 16 test documents, 160 questions and 800 choices. In CLEF 2012, Arabic was included for the first time in the

[6] http://www.clef-campaign.org/

QA4MRE as one of seven languages to be evaluated [44]. The test documents and reading tests were available in Arabic, Bulgarian, English, German, Italian, Romanian, and Spanish. The role of the participating systems is to select the most appropriate answer option. In the case of the system is not certain about the answer, it may leave some questions unanswered [20]. Three Arabic systems participated in this campaign: IDRAAQ [5] and Trigui et al's system. [116].

IDRAAQ is an Arabic QA system designed and implemented by Abouenour et al. [5]. The system composed of three modules: question analysis, passage retrieval and answer validation modules. The three modules are designed as shown in Figure 2.1. The designers tried to take benefit from the advantages provided by Arabic WordNet (AWN)[7] to enhance the quality of retrieved passages and thereafter the performance of the whole system.

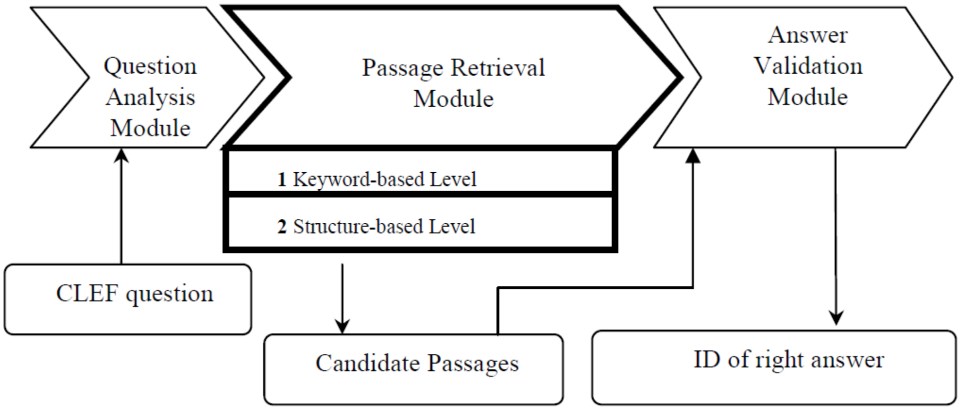


Figure 2.1: The system architecture of IDRAAQ [5]

In addition to the morphological query expansion, four semantic relations connecting AWN synsets were used. Namely: synonymy, hyponymy,

[7] http://globalwordnet.org/arabic-wordnet/

hypernymy and the SUMO[8]concept definition. Each keyword in the question is substituted by its semantically related words that are retrieved from the AWN. From each question, the query expansion process is applied only for keywords that are non stop-words. The QE component accepts as input a question keyword and for each keyword the system generates the following terms:

-Derivational forms and the root of the keyword using AL-KHALIL[9] system.

-Terms that share the same AWN synsets with the keyword including the super-types and the subtypes.

-Terms that share the AWN synsets that are hyponyms.

-Terms that share the AWN synsets that are hypernyms.

-Terms that related to AWN synsets provided by the SUMO concepts.

This process is repeated and a threshold is set in order to avoid endless recursive process. At the end of the process, for each question keyword, a list of words is generated. Each word is semantically related to the question keywords. Using these generated words, new queries will be formed by replacing each keyword in the question by its related terms. In the case of Named Entities keywords, the keyword is substituted only by its synonyms. To evaluate the system, two measures have been considered which are, accuracy and C@1 and two runs were conducted. The developers reported that the system reached 13% in accuracy and 21% in C@1 measure.

The second system was developed by Trigui et al. [116] which is based on information retrieval (IR) to deal with the problem. To find the best answer choice, the system retrieved the passages that have the question keywords and aligned them with the answer candidates, and then collected the answer included in these passages and selected it as the correct answer.

[8] http://www.adampease.org/OP/

[9] http://sourceforge.net/projects/alkhalil/

In case of there is no answer included in the passages, the system uses a list of inference rules deduced from the document collection to choose the answer. If after using the inference there is no answer founded in the retrieved passages, the question is leaved unanswered. Figure 2.2 shows the architecture of the system. The approach deals with only one type of questions, the non-complex questions. It tried to answer all the test-set questions and did not leave any questions as unanswered. The system obtained an overall accuracy and C@1 of 19%. The reason behind the poor performance is that the approach did not try to analyze the reading test document to answer the questions. It also depends on the background collection to offer enough redundancy for the passages retrieval, which made the system similar to the traditional question answering systems.

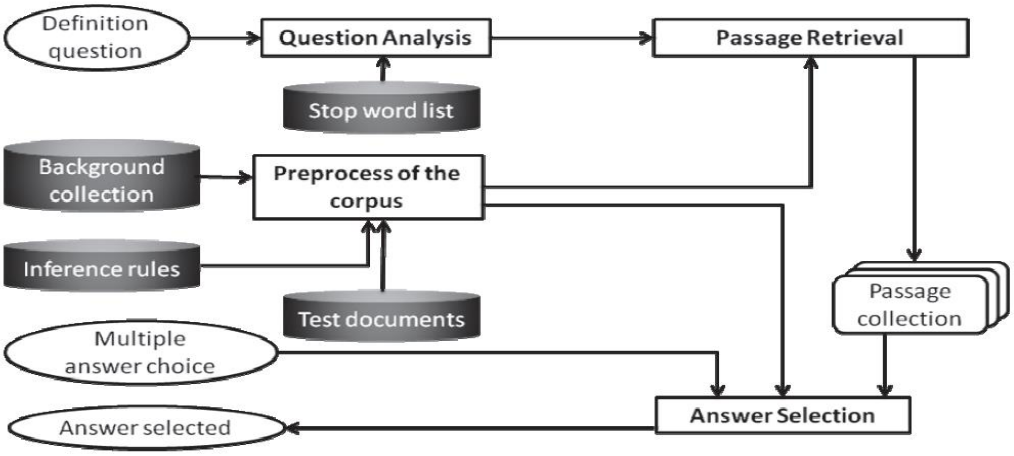


Figure 2.2: The system architecture of Trigui et al. [116]

Another QA system was developed by Ezzeldin et al. [47], they proposed a system to deal with comprehension reading question answering problem. The first version, which is named ALQASIM 1.0, participated in QA4MRE @ CLEF 2012. Their approach depends on answer keywords proximity to question keywords in the test document. It analysed the reading test text instead of the questions and scored the candidate answers according to three criteria: *(i)* the number of answer keywords found in the text within a distance threshold, *(ii)* the weights of all found keywords and *(iii)* the keywords distance from the question keywords. According to the authors, the first version achieved an accuracy of 31% and a C@1 of 36% without using any database collection tests.

In the second version, ALQASIM 2.0, the authors improved their system by utilizing three better techniques. These techniques are sentence splitting, background ontology semantic expansion and root expansion. Figure 2.3 shows the architecture of the system. They used sentence splitting as a natural boundary to search for answers in the test document. Since Arabic has a very rich and complex morphology, they expanded the keywords so the system can deal with different derivational forms of the words in the question answers and in the document. The expansion was applied to expand the document words with words from the same domain. In order to do that, they used an automatically generated ontology built from the CLEF 2012 background collections provided with the test-set.

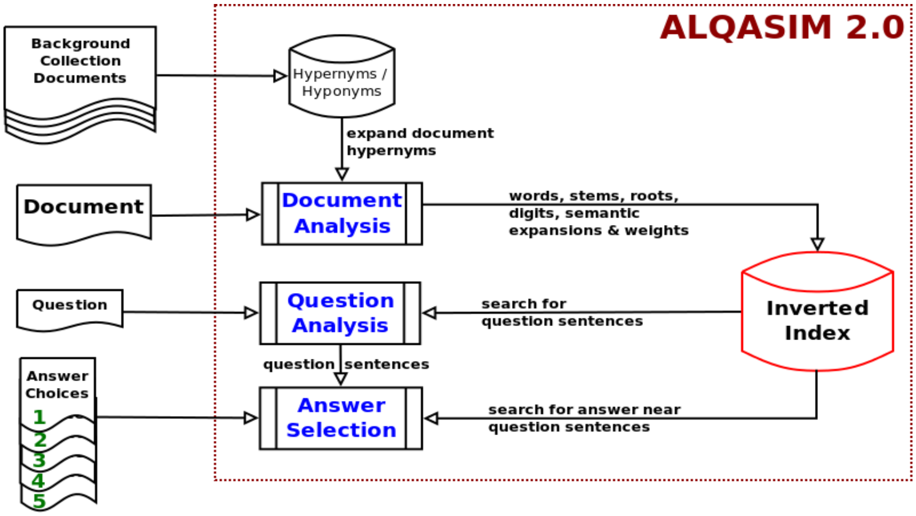

ALQASIM 2.0
Background Collection Documents
Hypernyms / Hyponyms
expand document hypernyms
Document
Document Analysis
words, stems, roots, digits, semantic expansions & weights
Question
Question Analysis
search for question sentences
Inverted Index
question sentences
Answer Choices
1
2
3
4
5
Answer Selection
search for answer near question sentences

Figure 2.3: The system architecture of ALQASIM 2.0 [47]

According to the authors, these techniques proved to be effective and led to a significant improvement in performance. The reported performance was an accuracy of 36% and a C@1 of 42% [47].

The performance of the Arabic systems was not very promising. However, it was a good initiative for research in the area of Arabic QA. On the other hand, regarding to non-Arabic systems, the best performing system, that had the highest score in the task of answer selection in QA4MRE campaign, was developed by Bhaskar et al. [98] to deal with English text. The system obtained the most promising results reaching an accuracy of 0.56 and C@1 of 0.65. The authors combined each candidate answer with the question in a hypothesis. They identified the query words from each hypothesis to retrieve the most relevant passages from the associated text. Each sentence was paired with the corresponding hypothesis and assigned a ranking score according to the textual entailment concept. The answer option that got the highest score among the list of candidate answers was selected as the correct answer. Figure 2.4 shows the architecture of the system.

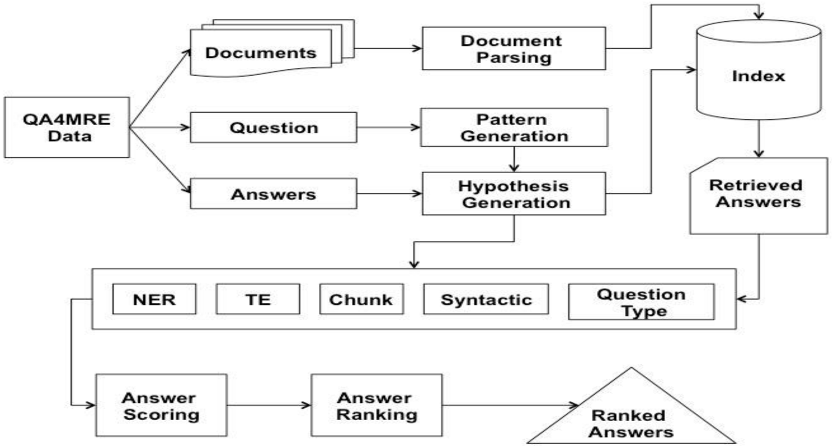


Figure 2.4: Answer Validation based Machine Reading System [98]

2.3 Textual Entailment Recognition in Arabic QA

ArbTE [16] was the first work to tackle the problem of recognising textual entailment in Modern Standard Arabic. The objective of ArbTE system was to assess the effectiveness of existing textual entailment approaches when they were applied to Arabic language. Given that recognizing textual entailment in Arabic is a non-trivial task and relies on the availability of accurate tools, the author combined the output of multiple data-driven dependency parsers and the output of three different taggers to get more accurate results in parsing and tagging respectively. After that, they utilized Tree Edit Distance (TED) algorithm to find the matching between two dependency trees of hypothesis and text pairs in Arabic. TED is one of the fast, simple and effective algorithms for finding the editing distance between ordered trees that was devolved by Zhang and Shasha in 1989 [122]. They extended the set of edit operations of standard TED algorithm to be applied to subtrees instead of only to single nodes. As per the author, both the strategies of combining different tools and the extension which made to deal with specific challenges posed by the language, have led to improvements over the performance of the task of textual entailment recognition in Arabic.

Khader et al. [70] also attempted to tackle the problem of Arabic textual entailment recognition. They adopted a lexical analysis method to assess the suitability of such methods for detecting textual entailment in Arabic. The system consists of two components: Preprocessing, Lexical and Semantic matching. The Preprocessing component contains three tasks: Part of Speech (PoS) Tagging, Stemming and Name Entity Recognition. The Lexical and Semantic matching component includes two steps: Firstly is to count the number of similar and synonyms words between each hypothesis and text pairs and secondly is to compute the bigram match between hypothesis and text. To evaluate the system performance, authors used

ArbTEDS[10] dataset which was developed by Alabbas [15], and compared their system with the human judgment. The system has reached precision of 68% for Entails and 58% for NotENtails with overall recall of 61%.

Mohammed and Mohammed [91] have studied the applicability of applying semantic similarity measures over Arabic WordNet. Seven semantic similarity measures were used. Three of them are linear path-based measures, namely, Wup, Path and LCH. Two measures are non-linear path-based measures LI and AWSS. The rest is one information content measure ResMeng and one is hybrid measure Zhou. For experiments with semantic similarity, the authors used AWSS benchmark, the Arabic dataset that was developed by Fazza et al. [50]. The results have been evaluated to assess the measures performance over AWN. The authors found that Wup measure achieved the best performance in similarity calculation compared to other measures while the worst performance was obtained by Path measure.

Almarwani and Diab [23] used distributional representations and traditional features in order to target the problem of Arabic TE without relying on any external resources in their work. They implemented multiple supervised frameworks using WEKA[11] software package. The set of the features they utilized to train their model is relatively small. These features are: Length, Similarity score, Named entity and Word embedding. The authors stated that using word representation based features resulted in good results compared to basic matching features. The logistic regression model achieved the best results among the used classifiers reaching an accuracy of 76.2 %.

Bakari et al [26] proposed an approach to recognize the textual entailment between the text and the question in the context of a question answering system. The method based on transforming the text and questions

[10] http://www.cs.man.ac.uk/~ramsay/ArabicTE/
[11] https://www.cs.waikato.ac.nz/ml/weka/

to logical predicates and then extracting the accurate answer. The method composed of five components: text analysis, question analysis, predicate generation, textual entailment recognition and answer generation. The algorithm starts with taking a text in html format as input and generates an annotated and analyzed text. The second step focuses on getting the possible reformulation of the questions that could be useful in the next steps of the answer generation. The next stage is the logical transformation where the text and the question are transformed to a set of logic predicates. Once the question and the text have been converted into logical forms, a list of entailments between the predicates of the question and the predicates of the sentences are recognized. Finally, the answer generation step which includes two tasks. The first is retrieving the candidate answers that have entailment to the user's query. The second is assigning scores to each of these sentences to produce a list of ordered answers according to their scores to choose the final answer.

EWAQ is an entailment metrics based Arabic QA system proposed by AL-Khawaldeh [21]. The system consists of three modules which are: Question Analysis, Passage Retrieval and Answer Extraction. It concentrated on improving the accuracy of Arabic *Why*-type questions through enhancing the process of re-ranking the passages that retrieved by search engines. The re-ranking process is achieved based on the degree of entailment similarity between the relevant retrieved passages and the questions. In order to increase the accuracy of the system's information retrieval, the author used AWN to identify all the possible words that have semantic relations in the question and passages. The system was evaluated using a dataset of 250 *Why* questions with their correct answers. The questions have been selected from five different fields (science, history, computer, politics and religion) by thirty Arabic native speakers. Yahoo, Ask and Google search engines were used to compare the system accuracy.

The author reported that the obtained results indicated that using entailment similarity in answer extraction is significantly helpful and the overall accuracy reached to 68.53%.

Chapter summary

In this chapter, we provided a review of related work. In this review we first introduced the advances in the history of Arabic QA systems giving more attention to answer selection task. After that we talked about QA4MRE @ CLEF[12] campaign where we described the participated Arabic systems and their results. Later on we discussed the utilization of Textual Entailment approaches in Arabic QA systems. In the next chapter, we describe the general architecture of QA systems that used by the QA community researchers.

[12] http://www.clef-campaign.org

Chapter 3

General Architecture of QA System

This chapter discusses the general architecture of QA systems. It describes the most common pipeline architecture that most of the QA systems share. This architecture is general and can be applied to any language. Later in this chapter, we provide a brief description of different categories of QA system, the appropriate evaluation metrics that used by the QA community researchers to assess and compare their work as well as the standard QA evaluation forums that are available to support evaluating different systems.

3.1 General Architecture of QA systems

Most of QA systems share a common pipeline architecture that consists of three distinct modules. The modules are: Question Processing, Passage Retrieval and Answer Processing. Each of these modules has a core component besides other supplementary components. Although most QA systems follow the common architecture, however they might have differences in the way they implement each subtask in the modules. Figure 3.1 shows the most common QA architecture.

3.1.1 Question Processing Module

Given a natural language question provided by the user as an input, the Question Processing module starts to process and analyze the question in order to create a useful representation of the required information for the next module [48]. This module usually consists of two components, namely: Question Classification and Question Reformulation [22].

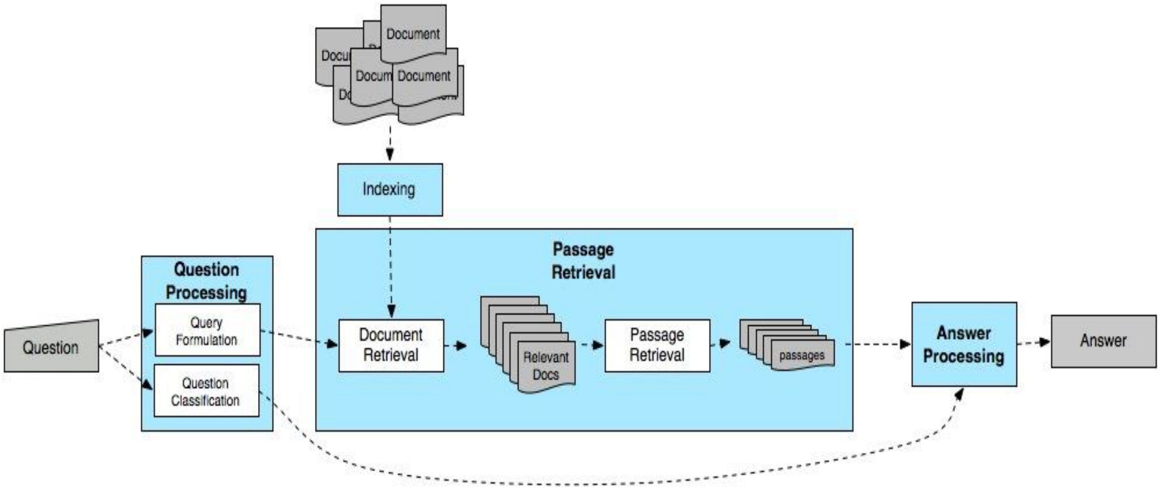


Figure 3.1: Question Answering system architecture [66]

3.1.1.1 Question Classification and Answer Type Detection

In order to get the right answer to the posed question, a question type classification process is performed to identify the question class. Knowing the type of the question helps the system limiting what kind of data is relevant, expecting the answer type (Entity), and developing answer patterns, which in turn leads to help next modules to locate and verify the answers correctly. The question is classified usually based on predefined categories of possible questions that are already coded into the system: what, why, who, how, when, where questions, etc [22]. After classifying the user question into one of these categories, the system predicts the type of entity expected to be found in the candidate answer sentences [103]. For example, a question like “What Libyan city has the largest population?” expects an answer of type CITY while a question like “Who founded British Airways?” expects an answer of type PERSON. Knowing the answer type for a question will help the system focusing on a specific entity rather than looking at every single sentence or noun phrase in the entire collection of documents [66]. QA systems usually consider the following entity types in the candidate answers: For factoid question, the entity type is expected to be location, percentage, date, organization, time, measure, monetary values, person or duration. For non-factoid QA systems, the answer type is expected to be reason or explanation [103]. Table 3.1 lists the question classes and corresponding expected answer types with examples. Li and Roth [84] proposed a hierarchical taxonomy in which questions were classified and expected answers were identified upon that taxonomy. Figure 3.2 shows the answer type taxonomy.

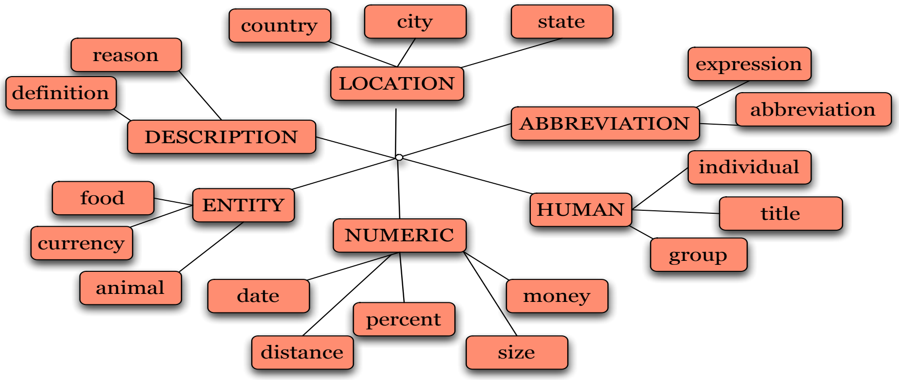


Figure 3.2: Answer type taxonomy [84]

In some cases, knowing the type of the question is not enough to find answers to all possible questions. This is because of some questions, such as, *what* questions are ambiguous in terms of what information is required to answer the question [62]. In order to deal with this issue, some systems extract something called a ***focus***. This can be performed by extracting a word or a sequence of words which indicates the main information that is required to answer the user's question [92][66]. For instance, the question

“What is the *highest building* in Canada?” has the focus “highest building”. Pattern matching rules based on the question type classification are used to accomplish this process [102]. If both the question type and the focus are known, then the system can more easily determine the type of answer required [22].

Table 3.1: Questions classification and the expected answer type

Question type	The expected type of answer	Example
مـتـى (mtY:When)	Date Time	مَتَى غَرِقَتْ تايتنكْ ؟ When did titanic sink?
أى (Ay: Which)	Location	اى مدينة لديها حرارة منخفضة؟ Which city has minimum temperature?
لماذا (lmA*A: Why)	Reason	لماذا ليس لدينا امطار كافية هذه السنة؟ Why don’t we have enough rain this year?
مـن (mn: Who)	Person, Organization	من هو رئيس تونس؟ Who is the president of Tunisia?
كـيـف (kyf: How)	Process	كيف يتم انتخاب رئيس الولايات المتحدة الأمريكية؟ How is the president of USA elected?
ايـن (Ayn: Where)	Location	أين تقع كينغستون؟ Where is Kingston located?
كم (km: How much, How many)	Numeric expressions	كم عدد المباني في هذا الشارع؟ How many buildings on this street?
مـا (mA: What)	City	ما هى عاصمة ليبيا ؟ What is the capital of Libya?

3.1.1.2 Query Reformulation

After identifying the “question focus” and “question type”, the next step is to extract a list of keywords from the remaining of the question to be passed to the Document Retrieval component in the Passage Retrieval module. Standard techniques such as NER, stop-word lists, and PoS taggers are applied to perform this process [22][78]. Each word is reduced to its morphological root using a rule-based stemmer or by looking up the morphological root in a machine readable dictionary [93]. The extracted keywords are sorted by their priorities, in the case of too many keywords are obtained from the query, then only the first N words are sent to the next stage [78].

Query Expansion

One of the important issues here is that most of the time users pose their questions using words which do not, necessary appear in the target documents. In fact, if documents contain the correct answer that does not include the whole or a part of the question keywords, they will not appear among the candidate passages, and as a result, the Answer Processing module will not be able to return a correct answer [7]. Therefore, in order to overcome such a problem, a Query Expansion (QE) process can be performed [8][4]. QE is one of the NLP techniques which can improve the quality of the IR component by expanding the list of keywords used to retrieve candidate passages. Expanding the user’s question keywords will help to generate new keywords that may exist in the target documents and not exist in the original query. The process of QE is classically performed on the basis of morphological relations. For example, if the user’s question includes the keyword معرفة (mErfp1, knowledge), the QE component can extend this keyword by providing its other morphological forms such as, the sound masculine plural يعرفون (EArfwn, they know), the present masculine verb يعرف (EArf, knows), the feminine subject عارفة (ArEfa, knower), and so on [6]. Query expansion in Arabic QA system can be improved by utilizing semantic web resources such as ontologies [103]. The most widely used general Arabic ontology is Arabic WordNet (AWN). Arabic WordNet offers alternate ways to expand a user input query by incorporating AWN in QA system, a more advanced QE process can be achieved depending on semantic relations between question keywords and document keywords [8]. Thus, additional semantically equivalent keywords can be added to the user query. For instance, if the system finds the keyword طريق (Tryq: a way) in the question posed by the user, in addition to expanding it to include more morphological forms like: طرق (Trq: broken plural of Tryq) or طرقات (TrqAt: other broken plural of Tryq), it also can be expanded at the semantic level to have other keywords like ممر (mmr : path) or مسار (msAr : trajectory) and so on, since they are similar in meaning with respect to the original keyword [4]. Few studies show that AWN can be used in Arabic QA system especially in QE to expand the user's query keywords, and subsequently enhance the passage retrieval task. Finally, the output of the former steps is a set of query terms those are ready to be passed from the Question Processing module to the Passage Retrieval module, which uses them to perform the IR process.

3.1.2 Passage Retrieval Module

The Passage Retrieval module in QA systems is also commonly referred to as Paragraph Indexing module [22][78]. This module is a core component of the QA system, where the reformulated question is submitted to the IR system, which in turn recognizes the documents that are estimated as relevant to involve the expected answer [8]. After identifying relevant documents, within the relevant documents, the module determines the passages most likely to contain the answer to the user query and retrieves them. The overall function of this module is to process the documents in order to retrieve a ranked list of relevant passages with the highest probability of containing the correct answer [22]. In order to do so, the Passage Retrieval module usually consists of three components: document retrieval, passage retrieval and passage ordering.

3.1.2.1 Document Retrieval

Generally, an IR system is used to retrieve documents and passages from a collection of document corpora. In the case of open domain QA, the system usually leverages an SE such as Google or Yahoo [78]. The task of the IR system in this phase is not to give the accurate answer to the user's question, but to identify and then retrieve a set of documents that contain the most representative words in the submitted question [18][22]. One of the most common techniques used in information retrieval to identify the documents relevant to the user's query is to create an inverted index of the knowledge base. By using an inverted index, we can find out what documents in the knowledge base contain a particular keyword in the user query. For example, if the user's question is :من هو رئيس الصومال؟ (Who is the president of Somalia?), the documents appearing in the inverted index of the words "رئيس" (president) and "الصومال" (Somalia), will be considered as relevant documents. The accuracy in recognizing the relevant documents is very crucial, as it will affect the performance of the passage retrieval phase and the answer extraction process.

3.1.2.2 Passage Retrieval

The main purpose of the passage retrieval or passage filtering is to decrease the number of candidate documents, and to decrease the amount of candidate text from each document. Since the number of retrieved documents by the IR system tends to be very large, these documents are generally filtered by Passage Retrieval component to exclude paragraphs that do not contain all the keywords of the query submitted by the user. The notion of passage retrieval is based on the principle that the most relevant documents should include the question keywords in a few adjacent passages, instead of scattered over the whole document [22][57][78]. Most of the passage retrieval techniques in the field of QA rely on this concept. In other words, a passage is considered more relevant if it contains a higher number of keywords with minimal distance between them [107]. The reasons behind shortening documents into passages in this step before processing them further in detail are: to make the QA system faster by processing less content since the response time of a system is very important, and to ensure that not a huge number of paragraphs are passed on to the next module [22].

3.1.2.3 Passage Ordering

After filtering out the passages returned by the IR that don't contain potential answers, the next important stage is the passage ordering stage, which sorts the extracted paragraphs to obtain a set of ranked passages according to a plausibility degree of containing the right answer. One of the approaches used for passage ordering is a pattern based approach. In this approach, a number of patterns for candidate answer sentences are used. The patterns are developed depending on the structure of the question and the possible answer type we expect to see in the answer. The passage containing these patterns and the entities of the right type is considered more relevant [103][66]. For example, if we have question, ما هى عاصمة ليبيا ؟ (What is the capital of Libya?), the candidate passages should contain sentences like, عاصمة ليبيا هى [مدينة] (The capital of Libya is [City]).

3.1.3 Answer Processing Module

As the final module in the architecture of QA system, the Answer Processing module which is responsible for identifying and extracting answers from the set of ranked paragraphs provided by the Passage Retrieval module [22][78].The Answer Processing phase consists of three major tasks are described as the followings.

3.1.3.1 Answer Identification

Taking in consideration the question type determined during the question processing process and the expected type of answer, the Answer Identification component tries to identify the candidate answers within the passages retrieved by the passage retrieval module. Since the answer type is not explicit in the question or the answer, parsing techniques such as NER are commonly used. Also, PoS tagger could be used in order to recognize the answer candidates within identified paragraphs [78][22]. After parsing the retrieved passages to recognize named entities (e.g. names of persons, organizations, dates etc.), the answer types returned by the parser are compared to the expected answer types derived in the question processing module. The outcome of this process is a set of candidate answers that could be ranked according to some algorithms [103].

3.1.3.2 Answer Extraction

The function of Answer Extraction component is to extract the answer by choosing only the word or phrase that answers the submitted query. After the recognition of the answer candidates performed by previous stage, a set of heuristics is applied in order to extract the correct answer from the answer candidates. Some of these heuristics can be defined based on number of keywords matched, distance between keywords, answer type match or other features [62][22][78].

3.1.3.3 Answer Validation

Before the answer is presented to the user, the answer validation step aims to validate the answer by assigning a score of confidence in the correctness of the answer. Given a question, a candidate answer and a support text, the answer validation determines if the specified answer is correct and supported or not [78][22][115]. The answer validation confidence score could be increased in a several ways. One way is to use a lexical resource to validate that a candidate response was of the correct answer type [78][22].

3.1.3.4 Answer Presentation

Finally, the system presents the answer to the user. Different QA systems use different approaches to present the answers. Some systems present a list of several ranked answers based on the appearance of the correct answer in the list. Some other systems are designed to choose and present only a single answer (the most likely answer) [78]. While other systems return URL links to provide users with some contextual information for the answers [124].

3.2 Classification of Question Answering Systems

There are different types of QA systems in the literature. In spite of the fact that most of these systems share general pipeline architecture, they vary from each other according to various dimensions. Next sections provide a brief description of different categories of Arabic QA system.

3.2.1 Domain Coverage Criteria

Based on the domains covered by them, QA systems can be classified into two categories: Closed-domain and Open-domain. Closed-domain QA systems deal with questions under a particular domain (for instance, business, law, medicine). The answers for user’s questions are searched within documents usually written by experts in the domain. Therefore, the quality of answers is expected to be high compared to the open-domain QA.

However, such QA systems are unable to give answers to questions out of the domain. The level of the user’s satisfaction usually depends on their domain knowledge [89]. AQAS (Mohammed et al., 1993) [90], is a restricted domain Arabic QA system. Open-domain QA systems treat with questions about nearly everything. This type of QA relies on world knowledge and general ontologies for generating answers to the user’s questions. Users do not need to have specific domain knowledge when using open-domain QA systems to formulate their queries. On the other hand, these systems deal with a large collection of data which would make the control of the quality of content not an easy task. Accordingly, the quality of generated answers is low [103]. The Arabic QA systems such as, ArabiQA (Benajiba et al., 2007) [28], QASAL (Brini, 2009) [32], and AQUASYS (Bekhti and Al-Harbi, 2011) [27] are considered as Open-domain QA systems.

3.2.2 Information Retrieval Approach Criteria

Different QA systems use different techniques to retrieve answers to the questions posed by the users. These techniques can be classified into two categories: statistical based approach and rule based approach. Statistical based approach is used by what is called data-driven Question Answering systems. These systems utilize large amount of data to apply statistical techniques, such as probability of relevance and similarity computation, in order to discover statistical relations between the questions and the documents and then retrieve answers [103]. ArabiQA (Benajiba et al., 2007) [28] is an example of statistical based QA. On the other hand, rule based QA systems focus on question analysis to determine expected answers. Predefined patterns are built for questions and answers. Identifying candidate passages is performed on the basis of matching of the predefined patterns. This can be done by applying information retrieval techniques and performing syntactic and semantic parsing for matching passages to extract answers to the given question. This approach does not require large training data, but building patterns for a natural language is a difficult task [89]. AQUASYS (Bekhti and Al-Harbi, 2011) [27] and QASAL (Brini, 2009) [32] are both rule based systems.

3.2.3 Language Supported Criteria

Based on language paradigm, QA system can be either monolingual or multilingual. In the Monolingual QA, the user's question, resource documents and system's answer are expressed in only one language. The documents are processed in language of the user's question without performing any type of language translation. Whilst, in multilingual QA systems, the user's question and resource documents are processed in different languages. In such systems, the user's question is translated into the languages of resource documents, the documents containing the expected answer are retrieved, then finally, the answer is returned to the user in the corresponding language. Different language processing techniques and translation tools are required to achieve the task. The accuracy of these tools is very crucial to avoid the risk of loss of concepts when translating questions because languages are usually different in lexical, syntax and rules. However, these multilingual QA systems are beneficial as information dispersed in different languages can be combined to get more knowledge [81] [89] [103]. An example of an Arabic monolingual QA is ArabiQA (Benajiba et al., 2007) [28].

3.3 Question Answering Systems Evaluation

Performance evaluation is a key to scientific progress. Evaluation of QA systems is a challenging task. It involves a large amount of manual effort. However, it is very important for QA systems to improve their performance [54]. Evaluation in English and other Latin-based languages have been receiving more interest than Arabic regarding question answering systems evaluation [109].

3.3.1 Question Answering Systems Evaluation Metrics

Using the appropriate evaluation metrics is necessary to help researchers to assess and compare the systems to measure the performances of different approaches. There are many evaluation metrics used in the area of QA. The following metrics are the most commonly used metrics in Arabic QA [109]:

Accuracy is used to evaluate the QA system performance in terms of its ability to retrieve relevant items and ignore irrelevant ones. It is calculated by dividing the number of relevant items retrieved plus the number of not relevant items that are not retrieved by the number of all items [109]. Table 3.2 presents the relationship between relevance and retrieval in the contingency matrix.

$$\text{Accuracy : Acc} = \frac{tp + tn}{tp + fp + tn + fn} \qquad (3.1)$$

Where:

tp: True Positives

tn: True Negatives

fp: False Positives

fn: False Negatives

Table 3.2: Information Retrieval contingency table

	Retrieved	Not Retrieved
Relevant	true positives	false negative
Not Relevant	false positives	true negative

Precision is defined as the number of relevant documents that are retrieved divided by the number of all retrieved documents.

$$\text{Precision : P} = \frac{tp}{tp + fp} \qquad (3.2)$$

Recall is defined as the number of relevant documents that are retrieved divided by the number of all relevant documents that exist in target collection.

$$\text{Recall: R} = \frac{tp}{tp + fn} \qquad (3.3)$$

F-measure is another metric was introduced in TREC evaluations. It is defined as a weighted harmonic mean of precision and recall. It trades off between precision and recall by using them together to provide a single measurement for a system. F-measure is computed by using the following equation.

$$F - measure\text{:} F = \frac{(1 + \beta^2)PR}{(\beta^2 P) + R} \qquad (3.4)$$

$$F\beta = 1 = \frac{2PR}{P + R} \qquad (3.5)$$

where β is a parameter indicating the importance of recall (R) and precision (P). The value of β controls the trade-off. When the value of recall and precision are equally important, β is assigned 1 [103].

Mean Reciprocal Rank (MRR) is another popular metric to measure the performance of QA systems. It was introduced in TREC 2001 QA track. It is a fractional number between 0 and 1 that indicates how many times the QA system ranks the correct answer as first. MRR is calculated based on two assumptions. First, the availability of a test-set of questions that are manually labelled with correct answers. Second, the system is designed to return a short ranked list of answers or passages that contain answers [66].

For example, if the system was designed to produce 5 possible answers to each question, then each question is scored according to the reciprocal of the rank of the first correct answer. For example, if the first correct answer of a question is in the third place, then the reciprocal rank value is 1/3. If the correct answer appears in the fourth place, then the reciprocal rank value will be 1/4. If the correct answer is the first one, then the reciprocal rank is 1/1 = 1. If there is no answer in the returned five answers, then the reciprocal rank value will be zero. MRR is the mean of all the questions' reciprocal rank values. The formula to calculate MRR of a QA system over *n* questions is defined as:

$$\mathrm{MRR} = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{rank(i)} \qquad (3.6)$$

MRR is more realistic for QA systems as it gives partial credit for answering a question correctly, but at lower rank.

3.3.2 Question Answering Evaluation Forums

For the performance evaluation of QA systems, there are a number of standard forums, available to support evaluating different information retrieval systems by testing them against a large text of data. Text REtrieval Conference (TREC) and, Conference and Labs of the Evaluation Forum (CLEF), both are widely accepted forums by the communities of QA researchers [103]. Text REtrieval Conference (TREC) is one of the annual conferences that established by the National Institute of Standards and Technology (NIST)[1]in 1992 in order to support research within the information retrieval community and to encourage the cooperation and

[1] http://trec.nist.gov/

technology transfer between different information retrieval research groups [119]. The track of QA began in 1999 with TREC-8. The first several editions of the track focused on factoid questions and systems were allowed to return 5 ranked answer snippets to each question. After that, the track has expanded both the type and difficulty of the questions asked. In 2002, the confidence-weighted score was introduced and systems had to return only one a single exact answer. The QA track of TREC-2003 included two tasks, the main task and the passage task. In the main task, in addition to factoid questions, list and definition questions are also considered. The passage task was similar to earlier Question Answering tracks. The test-bed consisted of a collection of documents (corpus of News articles) from different sources and a set of 500 fact based questions. From 2007 onwards, the QA track started to get step closer to IR rather than document retrieval by considering questions over blog documents besides the traditional documents. For many years, the QA track in TREC, has concentrated on the task of providing answers for human questions, but did not focus on real users from online community until 2015. In 2017, TREC started a new track called, LiveQA track, which focuses on real time questions that directly come from real users in real time. Subsequently, the task of QA became more realistic and much challenging [305]. The Conference and Labs of the Evaluation Forum (CLEF) is another evaluation forum that was launched in 2000. Its objective is to promote research in the field of cross-language systems. Since 2011, CLEF started Question Answering for Machine Reading Evaluation (QA4MRE). The QA4MRE task focuses on taking a single document and a set of questions as input and returning an exact answer as output. Questions are in the form of multiple choices and only one answer is correct [100].

Chapter Summary

In this chapter, we discussed the most common pipeline architecture that most of the QA systems share. The general architecture description is followed by a brief description of different categories of QA systems, the appropriate evaluation metrics that used by the QA community researchers to assess and compare their work as well as the standard QA evaluation forums that are available to support evaluating different systems.

Chapter 4

Arabic Challenges and Used Tools

This chapter is divided to two sections: The first section explains the importance of Arabic and how it differs from Indo-European languages. The significant challenges faced by researchers to build many natural language processing applications in general and QA specifically are discussed. The second section presents the main tools which have been used in this research.

4.1 Arabic Language Challenges

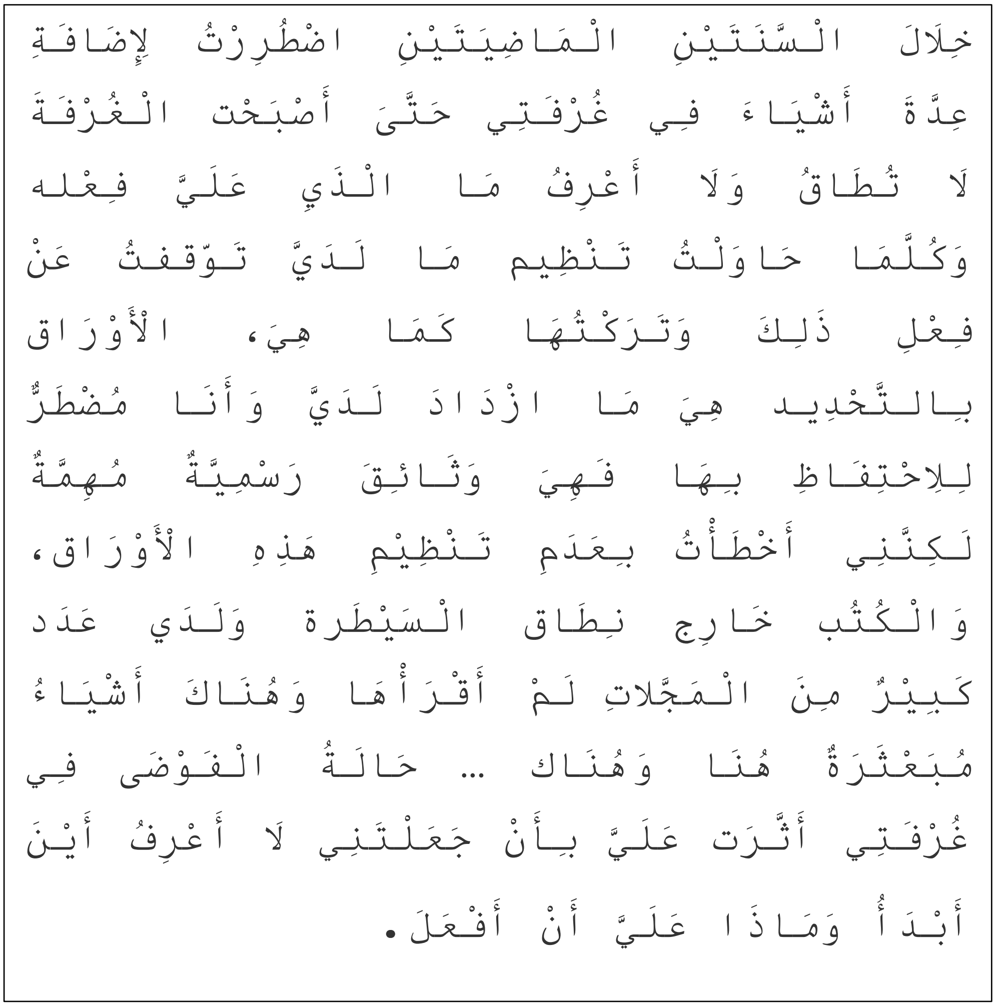
Arabic is a member of the Semitic languages family and the 6th most important language in the world with more than 250 million speakers. Moreover, Arabic is also used as a religious language by 1.5 billion Muslims around the world to perform their daily prayers regardless of their origin [12]. The modern form of Arabic is called Modern Standard Arabic (MSA) [108]. MSA is the official language of the Arab World, a region of 22 countries, and also the primary written language of the media and academic institutions. Arabic differs from Indo-European languages both syntactically and morphologically. The particularities of the Arabic language and the high level of complexity of its morphology and syntax, add extra complexities and challenges to the task of building a sophisticated Arabic QA system in comparison to other languages [61][49][46][29]. In fact, Arabic presents significant challenges to many natural language processing (NLP) applications in general and to QA specifically. In the following sub sections, some of these challenges are outlined.

4.1.1 Arabic Script

One of the key linguistic properties of the Arabic language that poses a challenge to the natural language processing is the Arabic script itself. The Arabic language has its own script, a right-to-left connected script that uses 28 basic letters (25 consonants and 3 long vowels). The shape of the character changes based on its location in the word (beginning, middle, end and separate). For example, the letter (غين) "'ghin" has an initial shape (غــ), a median shape (ــغــ), a final connecting shape (ــغ) , and a final non-connecting shape (غ). Most of the Arabic script letters are connected to other letters with few exceptional non-connective or right-connective letters. Fifteen of the twenty-eight Arabic letters contain dots to differentiate them from other letters. Figure 4.1 shows a sample of Arabic text.

There are other letters (or letter forms), namely different forms of hamza, often used in place of others due to varying orthographic conventions or common spelling and typing mistakes. These include [125][56]:

- "ى" (ya) and "ي " Yeh (alefmaqsoura).
- "ه" (ha) and "ة" (ta marbouta)
- "ا" (alef), "آ" (alefmaad), "أ" (alef with hamza above), and "إ" (alef with hamza below).
- "ء"(hamza), "ؤ" (hamza on w), and "ئ" (hamza on ya).



خِلَالَ الْسَّنَتَيْنِ الْمَاضِيَتَيْنِ اضْطُرِرْتُ لِإِضَافَةِ
عِدَّةَ أَشْيَاءَ فِي غُرْفَتِي حَتَّىَ أَصْبَحْت الْغُرْفَةَ
لَا تُطَاقُ وَلَا أَعْرِفُ مَا الْذَي عَلَيَّ فِعْله
وَكُلَّمَا حَاوَلْتُ تَنْظِيم مَا لَدَيَّ تَوّقفتُ عَنْ
فِعْلِ ذَلِكَ وَتَرَكْتُهَا كَمَا هِيَ، الْأَوْرَاق
بِالتَّحْدِيد هِيَ مَا ازْدَادَ لَدَيَّ وَأَنَا مُضْطَرٌّ
لِلِاحْتِفَاظِ بِهَا فَهِيَ وَثَائِقَ رَسْمِيَّةٌ مُهِمَّةٌ
لَكِنَّنِي أَخْطَأْتُ بِعَدَمِ تَنْظِيْمِ هَذِهِ الْأَوْرَاق،
وَالْكُتُب خَارِج نِطَاق الْسَيْطَرة وَلَدَي عَدَد
كَبِيْرٌ مِنَ الْمَجَّلاتِ لَمْ أَقْرَأْهَا وَهُنَاكَ أَشْيَاءُ
مُبَعْثَرَةٌ هُنَا وَهُنَاك ... حَالَةُ الْفَوْضَى فِي
غُرْفَتِي أَثَّرَت عَلَيَّ بِأَنْ جَعَلْتَنِي لَا أَعْرِفُ أَيْنَ
أَبْدَأُ وَمَاذَا عَلَيَّ أَنْ أَفْعَلَ.

Figure 4.1: Sample of Arabic text

This complexity of the Arabic orthography can confuse IR system. In these days, most of the existing NLP tools are developed for Latin languages and Arabic NLP researchers usually use some of these available tools in their works, but these tools are not completely suitable for the Arabic text, which represents many different difficulties to build Arabic QA [46].

4.1.2 Morphology

Morphology is one of the challenges facing developers in natural language processing (NLP). The morphology of any natural language is the linguistic system that governs how the words of this language are built. According to [24], “morphology refers to the branch of linguistics that deals with words, their internal structure, and how they are formed”. Due to its highly derivational and inflectional morphology, Arabic has a very rich and complex morphology which is called templatic or “root and pattern” morphology [82]. Derivational morphology concerns how words are formed and inflectional morphology concerns how words interact with the syntax [107]. Arabic is derivational because it is based on a root system to generate its words. Most of the nouns and verbs in Arabic are derived from a reduced number of roots (constant letters). Most of these roots consist of only 3 letters and few of them have four or five consonants. The derivations of words are formed by adding to each root one or more of the affixes (infix, prefix, and suffix) depending on around 120 patterns [69]. Table 4.1 shows an example of how three-grams root word “طلب” “Talba” (he requested) can be reformed to produce many words. Derivations in Arabic are usually templatic, thus we can say that: Lemma = Root + Pattern. The affixes can be added before, inside, or after a root, to generate more meaningful words [65]. Figure 4.2 shows an example of that.

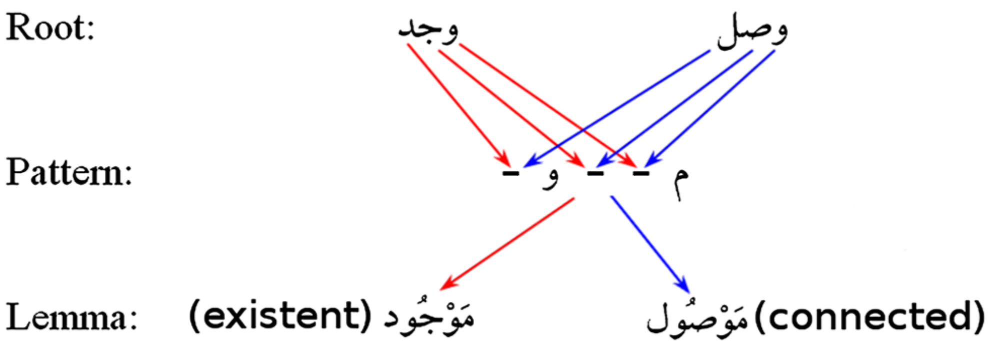


Figure 4.2: Example of Arabic derivation [46]

Arabic is also an inflectional language that takes the form of Word = prefix (es) + lemma + suffix (es). The prefixes can be articles, conjunctions or prepositions and the suffixes are objects or personal/possessive anaphora [29]. In Arabic, one word could replace a whole sentence in another language. For example, as shown in figure 4.3, the five-word sentence "and they will eat it" can be expressed in one word in Arabic "فسيأكلونها" which consist of, the stem "يأكل" (i.e. eat), the prefix "فس" (i.e., and will), the suffix "ون" (i.e. sound plural masculine pronoun) and the pronoun "ها" (i.e. singular object pronoun). These rich morphological features make the task of question analysis and query reformulation in Arabic QA much harder than other language [77].

4.1.3 Capitalization

Unlike English and other Latin-based languages, capitalization is not used in Arabic. Capital letters are very important to facilitate identifying proper names, acronyms, and abbreviations when it is supported in the objective language. Unfortunately, it is not the case for Arabic. For example, the Arabic word "أشرف " (Ashraf) could be used in a sentence as a given name

(proper name), a verb (supervised), or a superlative (the most honorable) [88][46]. Therefore, getting high performance in the task of named entity recognition is one of the obstacles that the Arabic QA faces [106].

Table 4.1: Examples of how the root “طلب” “Talba” can be reformed in Arabic [65]

Arabic word	Prefix	Infix	Suffix	Stem	Root	English Translation
الطالبين	ال	ا	يـن	طالب	طلب	Students (dual, masculine)
الطالبتين	ال	ا	تين	طالب	طلب	Students (dual, feminine)
الطالبان	ال	ا	ان	طالب	طلب	Students (dual, masculine)
الطالبتان	ال	ا	تان	طالب	طلب	Students (dual, feminine)
الطلاب	ال	ا	---	طلاب	طلب	Students (plural, masculine)
الطالبات	ال	ا	ات	طالب	طلب	Students (plural, feminine)
الطالب	ال	ا	---	طالب	طلب	Student (Singular, masculine)
الطالبه	ال	ا	ه	طالب	طلب	Student (Singular, feminine)
يطلب	ي	---	---	طلب	طلب	He requests (present tense, singular, masculine)
تطلب	ت	---	---	طلب	طلب	She requests (present tense, singular, feminine)

4.1.4 Broken Plural

The Arabic concept of “plural” is different from the English one. In English, a plural noun can refer to two or more of something. In Arabic, however, a plural noun refers to three or more of something. The plural in Arabic comes in two forms, the sound plural and the broken plural (BP). The formation of BP is more complex and often irregular. As an example, the plural form of the noun rjl (رجل , "man") is rjal (رجال , "men"), which is formed by inserting the infix alf (ا). But, the plural form of the noun ktAb (كتاب, "book") is ktb (كتب, "books"), which is formed by deleting the infix alf (ا). Thus, it is difficult to deal with Arabic BPs and reduce them to their associated singulars because no obvious rules exist, and there are no standard stemming algorithms that can process them [55].

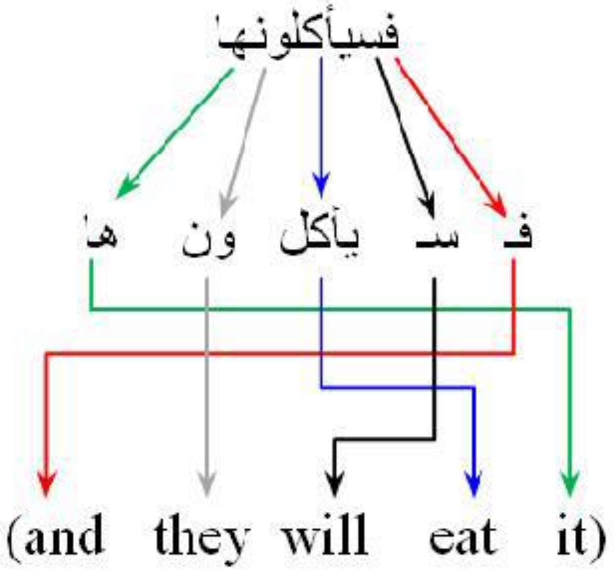


Figure 4.3: Example of Arabic Inflection **[**46**]**

4.1.5 Optional Short Vowels

Arabic script is characterized by diacritical marks (short vowels). By adding diacritics to words, the same word or phrase with different diacritics or with no diacritics can express different meanings [2]. Using diacritics can improve clarifying the context of a sentence or a paragraph [71]. However, in MSA, the diacritics are usually not written in a normal text like newspapers or a scientific book, whether in printed documents or digitized format. They are written only in some cases where the vowel marker is needed or in specialized contexts, such as children's books, dictionaries, and the Qur'an[1]. The omission of such diacritics in non-vocalized text also adds a lot of ambiguities to QA and IR applications due to the fact that an Arabic

[1] The holy book of Islam

word or a sentence represents different meanings with different diacritics [67]. For instance, the absence of diacritics in the phrase " كتب الولد فى المدرسة ", makes the phrase take at least two meanings, the first is: the books of a boy are in the school, and the second is: the boy wrote in the school. Likewise, the word (على) without vowels can mean the proper name (Ali) or the preposition (on). On the other hand, the word كَتَبَ and ,كتب might look similar to the eye, but to the computer, they do not match [43]. This level of ambiguity and vagueness presents a big challenge to Arabic QA and negatively affects the task of passages retrieval in the Answer Processing module when retrieving documents, passages and answers respectively especially for open domain Arabic QA systems that extract answers from the WWW where the content is rarely diacritized.

4.2 Used Tools

Different language processing resources and tools are required to achieve the task of answer selection. However, most of the existing NLP tools are developed for Latin languages. Arabic NLP researchers usually use some of these available tools in their works, but many of these tools are not completely suitable for Arabic. This is due to that languages are usually different in lexical, syntax and rules [46]. Several tools have been used in this research. Next sections present some of those tools.

4.2.1 Arabic WordNet (AWN)

Independently of the concerned language, WordNet (WN) is a large lexical database designed for use under program control [84]. Nouns, verbs, adjectives, and adverbs are grouped into sets of synonyms called synsets. A synset is a set of synonyms in a language that represent a single concept. Each word is represented by listing the word forms that can be used to express it. The synsets are interlinked by semantic relations such as, hyponymy, meronymy, antonymy...etc. The relations link between concepts not between words [77]. The first WN was built for the English language named Princeton WordNet (PWN). It includes most English nouns, verbs, adjectives and adverbs covering over 117,000 concepts (synsets) and over 150,000 English words [93]. WN is also mapped to SUMO[2] ontology. SUMO (Suggested Upper Merged Ontology) is a large formal public ontology used for research in linguistics applications and reasoning. It was created by merging publicly available domain ontologies into a one comprehensive structure. SUMO provides definitions for general terms and acts as a foundation for more specific domain ontologies. It contains about 1000 terms and 4000 definitional statements. SUMO is the only formal ontology that has been mapped to all of WNs. Figure 4.4 shows the mapping between SUMO and WNs [43].

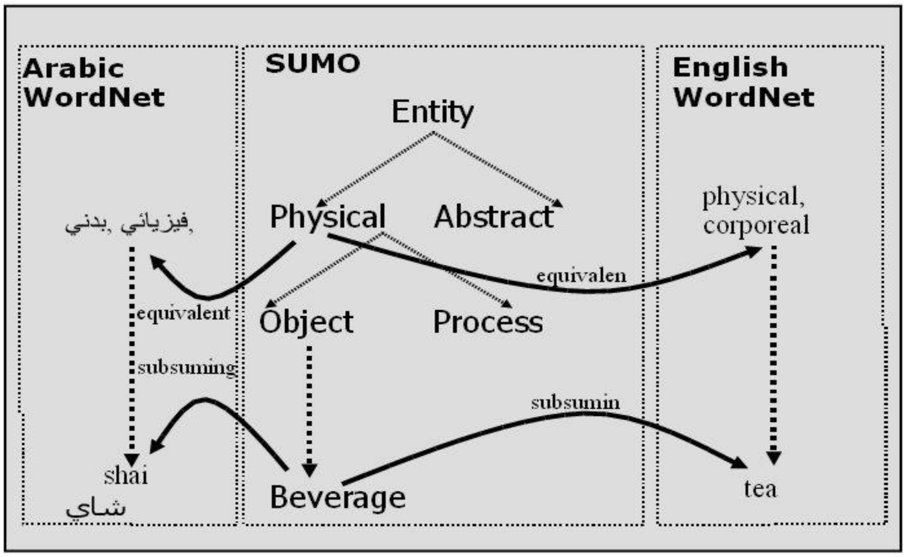
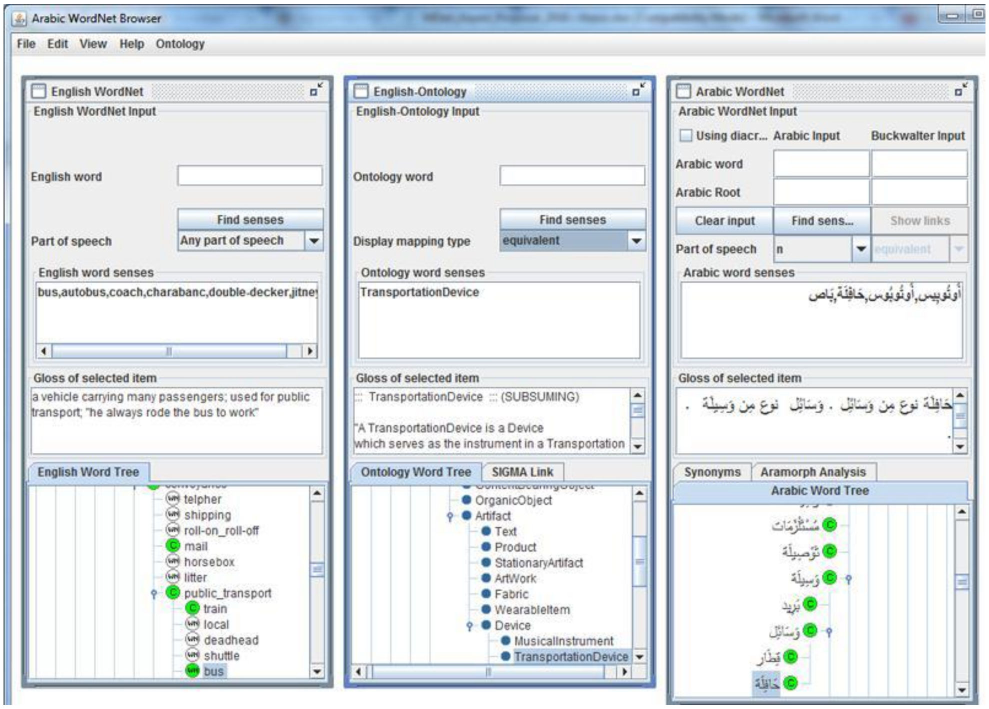


Figure 4.4: SUMO mapping to WordNet [43]

[2] http://www.ontologyportal.org

Arabic Word Net (AWN) is a freely available tool used as a lexical database for MSA. The first AWN was released in January of 2007. It is based on the widely accepted PWN for English. The construction of AWN followed the development process of English WN and Euro WN [32][79][46][43][55]. AWN is similar to its English counterpart WN in most of the aspects and the relations. It focuses on common-class words: nouns, verbs, adjectives, adverbs and adverbials. Each word can belong to one or several synsets. It has 11269 sunsets, 23481 words and 22 link types. Moreover, there is a direct mapping between word senses in AWN and those in PWN, enabling translation to English on the lexical level [29][44]. Figure 4.5 shows the Arabic WordNet browser interface. WordNet [53] is a very commonly used resource for discovering semantic relations between two fragments of text [51]. Few studies show that AWN can be used in Arabic QA system especially in Query Expansion (QE) to expand posed queries, and consecutively enhance passage retrieval task. The process of QE is classically performed on the basis of morphological relations. For example, if the user's question includes the keyword معرفة (mErfp1, knowledge) , the QE component can extend this keyword by providing its other morphological forms such as , the sound masculine plural يعرفون (EArfwn, they know), the present masculine verb يعرف (EArf, knows), the feminine subject عارفة (ArEfa, knower), and so on [6]. WordNet offers alternate ways to expand a user input query by incorporating WN in QA system, more advanced QE process can be achieved depending on semantic relations between question keywords and document keywords [8]. Thus, additional semantically equivalent keywords can be added to the user query. For instance, if the system finds the keyword طريق (Tryq: a way) in the question posed by the user, in addition to expanding it to include more morphological forms like: طرق (Trq: broken plural of Tryq) or طرقات (TrqAt: another broken plural of Tryq), it also will be expanded at the semantic level to have other keywords like ممر (mmr : path) or مسار (msAr : trajectory) and so on, since they are similar in meaning with respect to the original keyword [4]. In the task of textual entailment recognition (RTE), WordNet (WN) is one of the main resources that have been used widely to measure the similarity between the text (T) and the hypothesis (H) [34].



Part of speech: Any part of speech

English word senses: bus,autobus,coach,charabanc,double-decker,jitne

Gloss of selected item: a vehicle carrying many passengers; used for public transport; "he always rode the bus to work"

Display mapping type: equivalent

Ontology word senses: TransportationDevice

Gloss of selected item: :: TransportationDevice :: (SUBSUMING)

"A TransportationDevice is a Device which serves as the instrument in a Transportation

Part of speech: n

Arabic word senses: أُوتُوبِيس,أُوتُوبُوس,حَافِلَة,بَاص

Gloss of selected item: حَافِلَة نوع مِن وَسَائِل . وَسَائِل نوع مِن وَسِيلَة .

Figure 4.5: Arabic WordNet browser interface

4.2.2 The Farasa Arabic NLP Toolkit

Farasa[3] [105] is a state of the art open source toolkit that consists of several tools for Arabic text. Tools in Farasa were trained on the news that written in Modern Standard Arabic. The toolkit has been developed by Qatar

[3] http://qatsdemo.cloudapp.net/farasa/

Computing Research Institute (QCRI)[4] and made available to the research community. The Arabic NLP services offered by Farasa include:

4.2.2.1 Named Entity Recognition (NER)

NER is the task of identifying named entities, such as person names, places, organizations, monetary values, etc. in a raw text and classifies them into predefined categories. Due to the lack of capitalization and large knowledge bases in Arabic, getting high performance in the task of named entity recognition is one of the obstacles that Arabic researchers face. To address the problem of named entity recognition, the author combined cross-lingual features and knowledge bases from English using cross-lingual links. Three different features were utilized which are: Cross-lingual capitalization, Transliteration mining and DBpedia. For Arabic NER, the features led to improvements over a strong baseline system on a standard dataset [42].

4.2.2.2 Tokenization

The purpose of tokenization is to separate the text into single words (tokens). Farasa toolkit involves an Arabic segmenter that uses different features and lexicons in order to rank the possible segmentations of a word. The features are: prefixes, suffixes and their combination, underlying stem templates, likelihood of stems and presence in lexicons containing valid stems and named entities. The developers reported that the tokenizer outperforms some of the state of the art Arabic segmenters in terms of accuracy and efficiency [9].

4.2.2.3 Dependency parsing

Dependency parsing is the task of mapping a sentence to a dependency tree. The output of the dependency parsing is a tree where words are vertices and syntactic relations are dependency relations. Syntactic parsing aims to

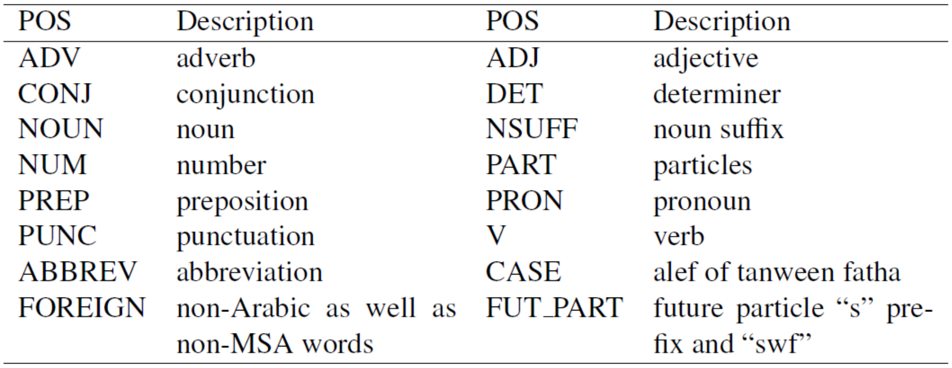
[4] https://qcri.qa/our-research/arabic-language-technologies

analyse sentences automatically, using the grammar rules in order to construct representations of their syntactic structure. Parsing in Arabic is non-trivial task due to its ambiguity. Therefore, the accuracy of dependency parsing in Arabic tends to be lower than a parsing in other languages. The authors developed their module based on randomized greedy algorithm that jointly predicts the tokenization, part of speech tags and the dependency parse. The algorithm greedily searches over a combination of parse trees and lattices that encode alternative morphological and POS analyses. It makes local modifications to part of speech tags and dependency trees iteratively [123].

4.2.2.4 Part of speech tagging (POS)

Part Of Speech (POS) tagger is natural language processing tool that used to assign a syntactic role for each word in a sentence depending on the way the word is used. Therefore, each word is determined and tagged as noun, verb, adjective, etc. Farasa part of speech tagger is designed to find the best tag for each clitic produced by the tokenizer in addition to determine the gender and number for each noun and adjective. A feature vector is built for each possible tag for each clitic. These vectors are fed to SVMRank to learn feature weights and then to assign a possible tag to each token. Table 4.2 shows the Part Of Speech tags of Farasa.

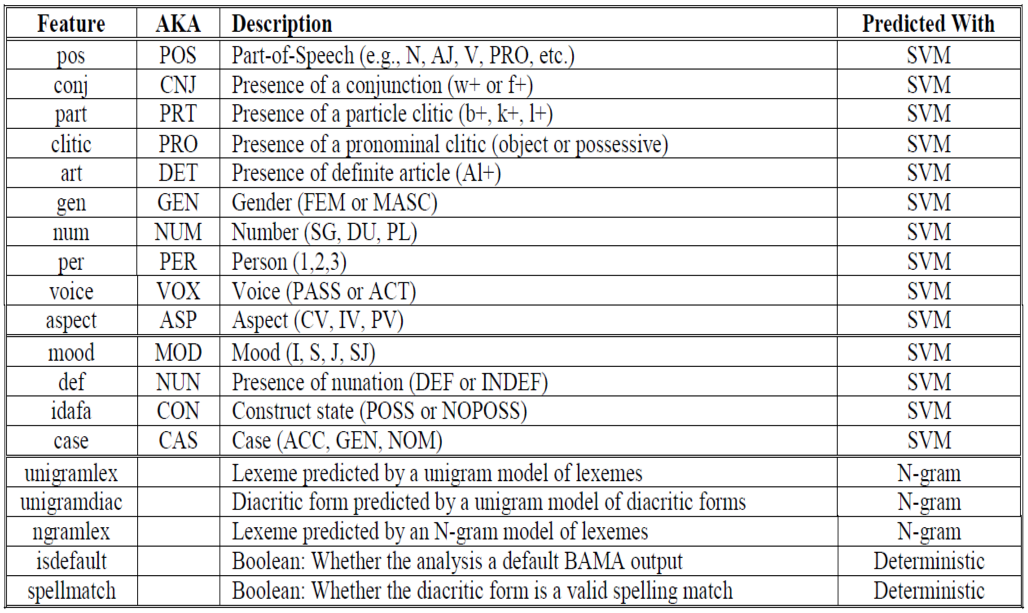
Table 4.2: Part Of Speech tags of Farasa [105].



POS	Description	POS	Description
ADV	adverb	ADJ	adjective
CONJ	conjunction	DET	determiner
NOUN	noun	NSUFF	noun suffix
NUM	number	PART	particles
PREP	preposition	PRON	pronoun
PUNC	punctuation	V	verb
ABBREV	abbreviation	CASE	alef of tanween fatha
FOREIGN	non-Arabic as well as non-MSA words	FUT_PART	future particle “s” prefix and “swf”

4.2.3 MADA+TOKAN Toolkit

MADA+TOKAN [59] is a freely available toolkit that provides one solution to different problems for Arabic NLP applications. It consists of two components. MADA and Tokan. MADA is a morphological analysis and disambiguation system for Arabic text. Given raw text, MADA tries to generate as much linguistic information as possible about each token in the input text. It applies support vector machine models and makes use of nineteen distinct, weighted morphological features in order to predict the best analysis that matches the current context. Table 4.3 presents the features used in MADA. TOKAN [58] is a general tokenizer for Arabic that takes the analysis produced by MADA as input to and generates a segmentation formatted to user specifications as output. The MADA system along with TOKAN utility provides an excellent toolkit for many Arabic natural language processing applications. Applications include stemming, diacritization, morphological disambiguation, POS tagging, glossing and lemmatization.



Feature	AKA	Description	Predicted With
pos	POS	Part-of-Speech (e.g., N, AJ, V, PRO, etc.)	SVM
conj	CNJ	Presence of a conjunction (w+ or f+)	SVM
part	PRT	Presence of a particle clitic (b+, k+, l+)	SVM
clitic	PRO	Presence of a pronominal clitic (object or possessive)	SVM
art	DET	Presence of definite article (Al+)	SVM
gen	GEN	Gender (FEM or MASC)	SVM
num	NUM	Number (SG, DU, PL)	SVM
per	PER	Person (1,2,3)	SVM
voice	VOX	Voice (PASS or ACT)	SVM
aspect	ASP	Aspect (CV, IV, PV)	SVM
mood	MOD	Mood (I, S, J, SJ)	SVM
def	NUN	Presence of nunation (DEF or INDEF)	SVM
idafa	CON	Construct state (POSS or NOPOSS)	SVM
case	CAS	Case (ACC, GEN, NOM)	SVM
unigramlex		Lexeme predicted by a unigram model of lexemes	N-gram
unigramdiac		Diacritic form predicted by a unigram model of diacritic forms	N-gram
ngramlex		Lexeme predicted by an N-gram model of lexemes	N-gram
isdefault		Boolean: Whether the analysis a default BAMA output	Deterministic
spellmatch		Boolean: Whether the diacritic form is a valid spelling match	Deterministic

Table 4.3: Features used in MADA [59].

4.2.4 ISRI Root Stemmer

Stemming a computational process used for to reducing words to their stems. A stem of a word is the part left after the affixes (prefixes, infixes and suffixes) have been removed. Information Science Research Institute's (ISRI) stemmer is an Arabic root stemmer developed by Taghva et al. [114]. Although, ISRI shares multiple features with the well know stemmer, Khoja stemmer [72] , however does not utilize a root dictionary for stemming. The lack of dictionary makes ISRI stemmer more capable to stem rare and new words, but on the other hand the extracted roots in some cases could be incorrect and useless for further tasks [45]. Stemming proceeds in the following steps:

1- Remove diacritics representing vowels.

2-Normalize the *Hamza* (ء) which appears in several distinct forms in combination with various letters to one form "أ".

3- Remove length three and length two prefixes respectively.

4- Remove connector "و" if it precedes a word beginning with "و".

5- Normalize " ا, آ, أ , إ " to "ا".

6- Return the stem if less than or equal to three.

7- Consider four cases depending on the length of the word.

Chapter Summary

In this chapter, we discussed the particularities of the Arabic language and the high level of complexity of its morphology and syntax. First, the significant challenges faced by researchers to build Arabic NLP applications were outlined. After that, we pinpointed the resources and tools are used to achieve our work.

Chapter 5

Features Engineering

Machine learning approaches are used by many researchers and different learning methods were applied to solve the problem of textual entailment since the introduction of the challenge, in RTE-1[94]. To solve the problem of recognizing the entailment between the text and the hypothesis, we modeled the textual entailment problem as a classification problem. This approach is considered more effective in case of the availability of training dataset [13]. Instead of using thresholds established by human experts, we utilized training data that were annotated in a way a classifier can read. Support Vector Machine (SVM) is known to achieve high performance and it proved to be very effective for a variety of natural language processing applications [87], and the most effective classifier in machine learning for classification tasks [13]. Therefore, in order to train and test our model for text entailment recognition, we used Support Vector Machine algorithm. SVM is a relatively new machine learning technique first presented by Vapnik [118]. Given a set of binary labeled training data, SVM algorithm maps the training data into a feature space of higher dimension, and seeks for the best hyperplane that separates all data points of one class from those of the other class, then optimizes that hyperplane for generalization. The best hyperplane for an SVM means the one with the largest margin between the two classes. The goal of SVM is to minimize the expectation of the output of sample error. Multiple variants of SVMs have been developed, we use a linear kernel SVM due to its popularity and high performance in classification problems in natural language processing applications compared to other kernels, especially when the number of features is high [13][110].

Let S= $\{(x_i, y_i)\}_{i=1..}$ be a set of training data, where $x_i \in R^d$ is a *feature vector* and $y_i \in \{-1,1\}$ are the class labels of x_i. If the corresponding label is +1, the x_i is called a positive instance; otherwise, it is a negative instance. The idea of SVM is to maximize the margin between the positive and negative instances. Margin is defined as the distance between the hyperplane and the training samples that are most close to the hyperplane. The support vectors (SV) are the data points that are closest to the separating hyperplane. These points are on the boundary of the slab and the hyperplane lies exactly in the middle of these support vectors. Figure 5.1 shows the maximum margins in SVM classification with its support vectors. All hyperplanes in R^d are parameterized by the weight vector (w) and the bias (b) which will be computed by SVM in the training process. The separator is defined as the set of points for which:

$$\mathbf{w}^{\mathbf{T}}\mathbf{x} + b=0$$

$$\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b > \mathbf{0} \quad \text{if } y_i = 1$$
$$\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b < \mathbf{0} \quad \text{if } y_i = -1 \qquad (5.1)$$

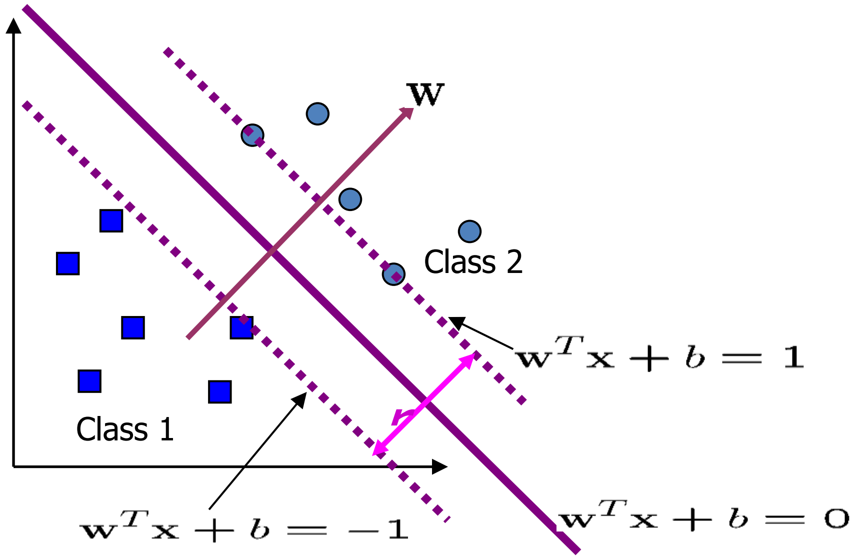
Our aim is to find such a hyperplane $f(x) = \text{sign}(w^T x_i + b)$, x:test data, that correctly classify our data.The distance from the hyperplane to a vector $\mathbf{x}_i$ is formulated as:

$$r = \frac{\mathbf{w}^T \mathbf{x}_i + b}{\|\mathbf{w}\|} \qquad (5.2)$$

Since our goal to maximize the margin, the decision boundary can be found by solving the following constrained optimization problem [39]:

$$\text{Minimize } \tfrac{1}{2}\,\|\mathbf{w}\|^2$$

$$\text{subject to} \quad y_i\,(\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b) \geq 1 \quad \text{for all } (\mathbf{x}_i, y_i),\ i=1..n \qquad (5.3)$$



w
Class 2
$\mathbf{w}^T\mathbf{x} + b = 1$
Class 1
$\mathbf{w}^T\mathbf{x} + b = -1$
$\mathbf{w}^T\mathbf{x} + b = 0$

Figure 5.1: Maximum margins in SVM classification.

In case of the set of the training data points are not linearly separable; the optimization problem cannot be solved. To deal with such case, soft margin SVM allows some data points to be mislabeled while still maximizing the margin. Slack variable ξ_i , is added, which will allow for noisy and outlying data points to violate the margins. Then the optimization problem can be solved as follows:

$$\textbf{Minimize } \tfrac{1}{2}\,\mathbf{w}^{\mathrm{T}}\mathbf{w} + C\sum_{i=1}^{n}\xi_i$$

$$\textbf{subject to } y_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b) \geq 1 - \xi_{i,} \ , \quad \xi_i \geq 0, i{=}1..n \qquad (5.4)$$

where C is a parameter to be tuned during training. $\xi_i \geq 1$, x_i is not on the correct side of the separating plane [39].

5.1 Used Features:

We use Support Vector Machine (SVM) to train our classifier and build the model file based on the selected feature sets. Each sentence (T) is paired with the corresponding hypothesis (H) to represent T-H pair. During the training stage, each T-H pair is represented by a feature vector (f_1,.., f_n) as input to the algorithm to induce the trained classifier. Then the classifier, during the classification stage, examines the features of unseen T-H pairs in order to classify them as entailed or not entailed pairs. Selecting appropriate features that give better results is the most significant part in machine learning. We combine three different sets of features that include syntactic, lexical and semantic features. The selected features are detailed in next sections.

5.1.1 Syntactic Features

The meaning of a sentence can be expressed with different word structures by different speakers. This means that two text pieces might have different lexical structure, but they have exactly the same meaning. When it comes to textual entailment recognition, that two semantically equivalence sentences with different lexical structure might not considered as entailed pair. Therefore, involving syntactic features is very important to deal with the issue of textual entailment. Syntax is “the study of the principles and processes by which sentences are constructed in particular languages” [38]. Given the importance of syntax in entailment recognition, syntax based methods are widely used in textual entailment recognition systems specifically in English [60][97][111][120]. Syntactic information in RTE has been used in several ways. Some of methods used shallow parsing approaches. POS tagger is used to assign a syntactic role for each token in the sentence in order to determine whether a word is a noun, verb, etc., while other methods utilized full syntactic analysis techniques to measure the similarity between two sentences.

5.1.1.1 Syntactic Parsing

Syntactic parsing aims to analyse sentences automatically, using the grammar rules in order to construct representations of their syntactic structure. One of the representations has been proposed by researchers to achieve this goal is dependency structure grammar.

5.1.1.2 Dependency Parsing

Since the meaning of a text fragment is not only based on the meaning of its words but also on the relations between these words, one of the important steps to recognize the entailment relation between two sentences is to consider the connections between the words in the sentences. In order to investigate the syntactic similarity between the Text (T) and the Hypothesis (H), we transformed both T and H from natural language sentences into syntactic graphs using Arabic syntactic parser. Farasa text processing toolkit for Arabic text[1] is used to achieve the task.

The output of the dependency parsing is a tree where words are vertices and syntactic relations are dependency relations. Figure 5.2 shows the dependency-parsing graph of the sentence (`` تنتشر الكثير من الاوبئة التى تهدد حياة الملايين في افريقيا ``: *Many epidemics threaten the lives of millions in Africa*) as an example. Table 5.1 shows types of dependency relations between the words in the above sentence after processed by the Arabic parser.

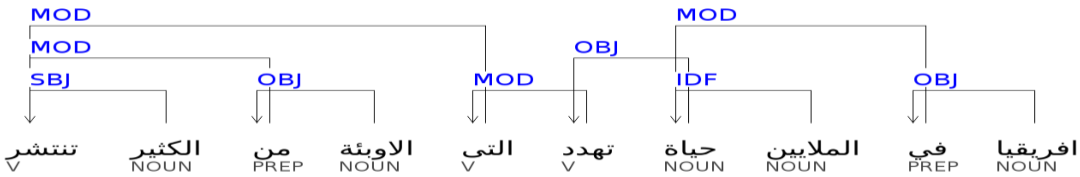


Figure 5.2: The dependency-parsing graph of the sentence (`` *تنتشر الكثير من الاوبئة التى تهدد حياة* الملايين في افريقيا``: *Many epidemics threaten the lives of millions in Africa*)

We can see that the above graph contains several kinds of dependency parsing relation, such as subject-verb relation SBJ (الكثير, تنتشر) (Alkvyr: many, tnt$r: spread) and object-verb relation OBJ (تهدد, حياة) (thdd: threaten, HyAp: life).

[1] http://qatsdemo.cloudapp.net/farasa/

Table 5.1: The dependency relations between words of the sentence (`` *تنتشر الكثير من الاوبئة التى تهدد* حياة الملايين في افريقيا``: *Many epidemics threaten the lives of millions in Africa*)

Dependency Relation	HEAD ID	FORM	ID
--	0	تنتشر	1
SBJ	1	الكثير	2
MOD	1	من	3
OBJ	3	الاوبئة	4
MOD	1	التى	5
MOD	5	تهدد	6
OBJ	6	حياة	7
IDF	7	الملايين	8
MOD	7	في	9
OBJ	9	افريقيا	10

5.1.1.3 Selected Syntactic Features

Syntactic relations are different in their importance in entailment recognition [37]. In our system, we try to use the most important syntactic features that could give us a better indication for the textual entailment recognition. Four syntactic features were used, the features are: Subject – Verb, Object – Verb, Subject – Subject, Object – Object.

A) Subject-Verb

This feature bases on the common subjects and the verbs between each T-H pair that have the *SBJ* relation. It calculates the ratio of the count of matched subjects and verbs (that tagged by the parser as *SBJ* through the dependency relation identification) between each text (T) and the corresponding hypothesis (H) to the count of subjects and verbs in hypothesis (H). The feature is defined as follows:

$$SBJ\ (T,H) = \frac{number\ of\ matched\ SBJs\ between\ H\ and\ T}{Total\ number\ of\ SBJ\ in\ H} \qquad (5.5)$$

B) Object-Verb

Object-Verb feature is concerned with the shared verbs and objects between each text and its hypothesis that have OBJ relation. The feature is calculated as the ratio of the count of shared objects and verbs (that tagged by the parser as *OBJ* through the dependency relation identification) between each text (T) and the corresponding hypothesis (H) to the count of subjects and verbs in hypothesis (H). The feature is defined as follows:

$$OBJ\ (T,H) = \frac{number\ of\ matched\ OBJs\ between\ H\ and\ T}{Total\ number\ of\ OBJ\ in\ H} \qquad (5.6)$$

C) Subject-Subject

This feature bases on the calculation of the ratio of the common subjects matched between the text (T) and the hypothesis (H) to the number of subjects in the hypothesis. The feature is computed as follows:

$$\text{Subject match} = \frac{number\ of\ shared\ subjects\ between\ H\ and\ T}{total\ number\ of\ subjects\ in\ H} \qquad (5.7)$$

D) Object-Object

The feature is defined as the ratio of the matched objects between each hypothesis (H) and the supporting text (T) to the number of objects in the hypothesis (H). The feature is calculated as follows:

$$\text{Objects match} = \frac{\#\ of\ shared\ \text{objects}\ between\ H\ and\ T}{total\ number\ of\ \text{objects}\ in\ H} \qquad (5.8)$$

5.1.2 Semantic Features

Six semantic features are used; three of them are Lexical Semantic features namely: Synonymy, Hyponym and Hypernym matching. The others are Semantic Similarity features, namely, WuP, Path and LCH metrics.

5.1.2.1 Lexical Semantic Features

During the process of lexical matching between the text and the hypothesis, there may be some words in the text (T) do not appear in the hypothesis (H). In order to discover the semantic relation between H-T pairs, we utilized Arabic WordNet (AWN) [31]. In Arabic WordNet, nouns, verbs, adjectives, and adverbs are grouped into sets of synonyms called synsets. A synset is a set of synonyms in a language that represent a single concept. Each word is represented by listing the word forms that can be used to express it. The synsets are interlinked by semantic relations such as, hyponymy, meronymy, antonymy…etc. The relations link between concepts not between words [77]. We utilize three types of matching, synonymy, hyponymy, hypernymy. A synonym is "a word or phrase that means exactly or nearly the same as another word or phrase in the same language"[2]. For example, a car *is an* auto. In linguistics, hyponyms and hypernym show the relationship between a generic term (hypernym) and a specific instance of it (hyponym). A hyponym shares a type-of relationship with its hypernym. Hyponym is a word with a more specific meaning than a general term. For example, car *is a hyponym of* vehicle. Hypernym is a word with a broader meaning than a specific term[3]. For example, vehicle *is a hypernym of* car. Figure 5.3 shows an example of the relationship between hyponyms and hypernym.

[2] https://en.wikipedia.org/wiki/Synonym

[3] https://en.wikipedia.org/wiki/Hyponymy_and_hypernymy

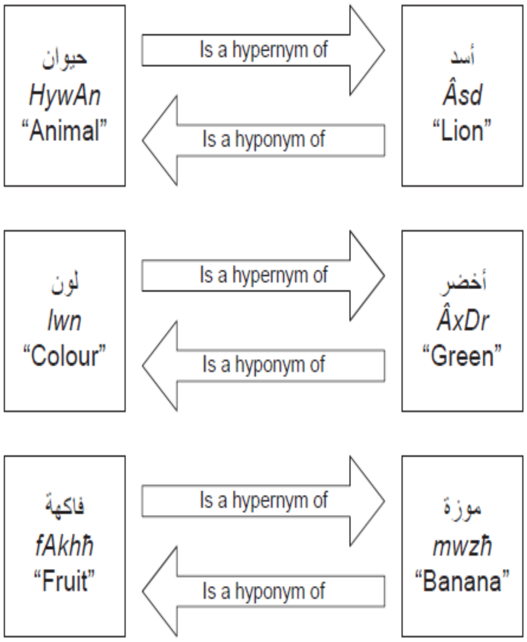


Figure 5.3: An example of the relationship between hyponyms and hypernym

A) Synonymy Matching

The lexical unit T entails the lexical unit H if they can be synonyms according to WordNet or if there is a relation of similarity between them [63]. All the nouns, verbs and adjectives from that are non stop-words are checked for synonyms. The following example shows the semantic entailment between some words in the Text and the Hypothesis:

Text

بفضل النشطاء في مجال الصحة العامة والعدل الاجتماعي في أفريقيا ، يزداد عدد الذين يعرفون
أن أفريقيا جنوب الصحراء هي الآن المركز الأساسي لكثير من الامراض: ثلاثة ارباع الذين
يموتون بسبب مرض الايدز في العالم هم من افريقيا....

Thanks to the activists in public-health and social-justice in Africa, growing numbers of people around the world know that sub-Saharan Africa is the epicenter of many diseases: three-quarters of AIDS deaths worldwide have been in Africa…

Hypothesis

تنتشر الكثير من الاوبئة التى تهدد حياة الملايين في افريقيا

Many epidemics threaten the lives of millions in Africa

AWN has a synset: “وباء,مرض ” (wbA': epidemic, mrD: Illness) which contains the two words as a synonyms, therefore, the two words are considered as entailed words. Synonymy matching feature calculates the overlap between the words in the hypothesis (H) that match words synonyms in the corresponding text (T) based on AWN. The measure is defined by the following equation:

$$\text{SynMatch} = \frac{\#\ of\ common\ synonyms\ between\ H\ and\ T}{\text{total number of words in H}} \qquad (5.9)$$

B) Hyponym and Hypernym Matching

A token *A* entails a token *B* if there is a path from one synset of *A* to one synset of *B* with hyponymy and / or entailment relations between intermediate synsets [63]. For nouns, *B* is a hypernym of *A* if every *A* is a (kind of) *B* and *B* is a hyponym of *A* if every *B* is a (kind of) *A*. For verbs, a verb *B* is a hypernym of a verb *A* if the activity *A* is a (kind of) *B*. Next examples show the hypernym entailment between the Text and the Hypothesis:

Text

كل شخص يرتكب جريمة أو يحمل أي نوع من الأسلحة سيواجه اتهامات بحيازة ذلك السلاح

Every person commits an offence or carries any kind of weapons will face charges for the possession of that weapon.

Hypothesis

اى شخص يحمل مسدسا سيواجه اتهامات بحيازة سلاح

Anyone who carries a gun will face charges of possessing a weapon.

According to the AWN hierarchy, the word "مسدس" (msds: Gun) is a kind of the word "سلاح" (silAH: Weapon). This means that "مسدس" (msds: Gun) *is a hyponym* (subtype) *of* "سلاح" (silAH: Weapon). The following example shows the hypernym entailment between the Text and the Hypothesis:

Text

الكلاب تحرس اصحابها وتحميهم من الغرباء

Dogs guard their owners and protect them from strangers.

Hypothesis

الكلاب حقا تهتم بأصحابها

Dogs really care about their owners

By considering AWN hierarchy, the verb "اهتم" (Ahtm: To care) *is a hypernym* (super_type) *of* the verb "حرس" (Hrs: To guard). This means that the words "اهتم" (Ahtm: To care) and verb "حرس" (Hrs: To guard) are entailed words. To calculate the features, each word in the Text (T) - Hypothesis (H) pair is checked for hyponyms and hypernyms. Hyponymy and hypernym matching feature is based on the overlap between number of hypothesis words that are hyponyms and hypernyms of other words in the text and the total number of the hypothesis words. The feature is defined by the following equation:

$$\text{hyponyms/hypernyms Match} = \frac{\text{\# of hyponyms / hypernyms between H and T}}{\text{total number of words in H}} \quad (5.10)$$

5.1.2.2 Semantic Similarity Features

There are several semantic similarity measures that have been developed with the purpose of quantifying how much two concepts are alike [34]. These similarity measures are based on the word to word similarity metrics [35]. They used to compute the similarity between two words at the semantic level, without taking their respective contexts into consideration [121]. A lexical data base such as WordNet is used to calculate the semantic similarity between a text (T) and a hypothesis (H). Semantic similarity metrics cannot be calculated for all parts of speech in the sentence since some of these metrics depend on the calculation of the information content values for the word sense, or some parts of speech do not appear in WordNet, such as proper nouns [11]. These metrics are particularly limited to verb-verb and noun-noun pairs since adjectives and adverbs are not classified into is-a hierarchy in WordNet [101]. Semantic similarity measures are classified into four categories: Information content-based measures, Path-based measures, Feature-based measures and Hybrid measures [112].

In English, there are many semantic similarity measures that have been used to calculate the similarity between the T- H pairs based on WordNet. However, very limited studies have been concerned with investigating the impacts of these measures on Arabic [91]. We utilize AWN to apply Path-based measures in order to measure the relatedness between the words in the T-H pairs and calculate the similarity between them as a result. The following semantic similarity metrics that we use in our work are based on path lengths between a pair of concepts in AWN:

A) WuP

Wu & Palmer (WuP) is a semantic measure was presented by Wu & Palmer [121]. WUP calculates the similarity between two words by considering the length between two given synsets S_1 and S_2 in the WordNet taxonomy as well as the distance between the LCS (least common subsumer) and the root of the taxonomy in which the synsets reside. WUP Similarity is defined as follows:

$$\mathrm{Sim}\boldsymbol{WUP}\,(\mathrm{S}_1, \mathrm{S}_2) = \frac{2 \times \mathrm{depth}\big(LCS\,(\mathrm{S}_1, \mathrm{S}_2)\big)}{\mathrm{depth}(\mathrm{S}_1) + \mathrm{depth}(\mathrm{S}_2)} \qquad (5.11)$$

where S_1 and S_2: are the synsets to which the words being compared belong. LCS $(\mathrm{S}_1, \mathrm{S}_2)$: is the least common subsumer of S_1 and S_2.depth(S_1): is the shortest distance from root node to a node S_1 on the taxonomy. depth(LCS(S_1, S_2)) is the length between LCS of S_1 and S_2 and the root of taxonomy.

B) Path

Path is a simple semantic measure that uses the path length distance to measure the similarity between two concepts in WordNet. The distance between concepts is computed by counting the nodes (sunsets) in the path [34]. Path measure is equal to the inverse of the shortest path length between two synsets in WordNet [101]. Path Similarity is defined as follows:

$$\mathrm{Sim}\boldsymbol{Path}\,(S_1, S_2) = \frac{1}{distnode(S_1, S_2)} \qquad (5.12)$$

Where $distnode(S_1, S_2)$: is the distance between synset S_1 and synset S_2 using node counting.

C) LCH

Leacock & Chodorow (LCH) is a path-based metric was presented by Leacock & Chodorow [80]. In order to measure the similarity, LCH finds the length between $S_1$and S_2 and maximum depth of the taxonomy in which $S_1$and S_2 are located. LCH similarity is calculated as follows:

$$\mathrm{Sim}\boldsymbol{LCH}\ (S_1, S_2) = -\log\ (\ \frac{\mathrm{len}(S_1, S_2)}{2\mathrm{D}})\qquad (5.13)$$

where D : is the maximum depth of S_i in the taxonomy (considering only nouns and verbs) and len: is the length of the shortest path between the two synsets.

5.1.3 Lexical Features

Seven lexical features are considered, three of them are N-gram overlap features namely, Unigram, Bigram and Trigram matching. The remaining features are: Longest Common Subsequence (LCS), Named Entity matching, Cosine similarity and POS matching.

5.1.3.1 N-gram Overlap Features

N-gram overlap is one of basic ways of recognizing the entailment relation between any kinds of text fragments. It can be measured by the counting the number of words they share. In our system, we calculate the percentage of N-gram word overlap between the supports text (T) and the corresponding hypothesis (H). The idea behind this heuristic is that the more shared words between the text (T) and the hypothesis (H), the more likely that H entails T, and vice versa.

Three N-gram features are adopted in our system: Unigram matching feature that measures the percentage of words of hypothesis (H) in the text (T); Bigram matching feature which calculates the percentage of bigrams (pairs of adjacent words) of hypothesis (H) in the text (T); and Trigram matching feature that compute the percentage of the trigrams of hypothesis (H) present in the text (T). All functional words are already eliminated during the stemming process in the keyword identification phase. Therefore only non-stop-words are considered for matching.

A) Unigram Matching

Each Text-Hypothesis pair is checked to calculate the number of the similar words appear in both of them. Next example shows the overlap between the Hypothesis and the Text.

Text

ومن الأسباب الغير معروفة **المؤدية** إلى زيادة الاحتباس الحراري هو **استخدامنا** السيء
للمياه والتدخل في مساراتها

One of the unknown causes of increasing global warming is our abuse of water and interference in its pathways

Hypothesis

الاستخدام السيء للمياه **يؤدى** إلى زيادة الاحتباس الحراري

The abuse of water leads to increasing global warming

From the above example, the number of common unigrams between the text (T) and the hypothesis (H) is two, which are: “ماء” (mA': water) and “ادى” (AdY: leads to). This process calculates the ratio of the count of shared unigrams between each text (T) and the corresponding hypothesis (H) to the count of unigrams in hypothesis (H). In order to calculate the unigram match we use the following equation:

$$UM = \frac{number\ of\ shared\ unigrams\ between\ H\ and\ T}{total\ number\ of\ unigrams\ in\ H} \qquad (5.14)$$

B) Bigram Matching

The number of shared bigrams (pairs of adjacent words) between the retrieved sentence (T) and the associated hypothesis (H) are calculated. The feature calculates the ratio of the count of shared bigrams between each text (T) and the corresponding hypothesis (H) to the count of bigrams in hypothesis (H). If we look at the same H–T pair below:

Text

ومن الأسباب الغير معروفة المؤدية إلى زيادة الاحتباس الحراري هو **استخدامنا السيء** للمياه والتدخل في مساراتها

One of the unknown causes of increasing global warming is our abuse of water and interference in its pathways

Hypothesis

الاستخدام السيء للمياه يؤدى إلى زيادة الاحتباس الحراري

The abuse of water leads to increasing global warming

We find that there is one bigram exist in both the Text and the Hypothesis, which is “الاستخدام السيء” (AlAstxdAm Alsy': the poor use). The Bigram matching is computed as follows:

$$BM = \frac{number\ of\ shared\ bigrams\ between\ H\ and\ T}{total\ number\ of\ bigrams\ in\ H} \qquad (5.15)$$

C) Trigram Matching

Trigram match feature (triple of adjacent words) is similar to the unigram and bigram features. Each text (T) and support hypothesis (H) pair is checked to calculate the number of the trigrams words appeared in both of them. From the same above example, we can extract one trigram shared between the text (T) and the hypothesis (H) which is: “الحراري زيادة الاحتباس” (zyAdp AlAHtbAs: increasing of thermal retention). The following equation is used to calculate the Trigram matching:

$$TrM = \frac{number\ of\ shared\ trigrams\ between\ H\ and\ T}{total\ number\ of\ trigrams\ in\ H} \qquad (5.16)$$

5.1.3.2 Longest Common Subsequence (LCS)

Longest Common Subsequence (LCS) is one of the effective features to compare the similarity of two sentences [104]. It measures the similarity between a text (T) with length *n* and a hypothesis (H) with length *m*, by searching in-sequence matches that reflect sentence level word order [74]. Formally, A string $\boldsymbol{A}$ = [a_1, a_2, ..., a_n] is a subsequence of another string $\boldsymbol{B}$ = [b_1, b_2, ..., b_m], if there is a strict increasing sequence [i_1, i_2, ..., i_k] of indices of $\boldsymbol{B}$ such that for all **j** = *1, 2, ..., k*, we have $\boldsymbol{b}_{ij}$ = $\boldsymbol{a}_j$. Given T-H pair, the longest common subsequence (LCS) of *T* and *H* is a common subsequence with maximum length [83]. The idea behind that is the longer the LCS of the Text-Hypothesis pair is, the more similar the text (T) and the hypothesis (H) are [75]. The LCS feature is described as follows:

$$LCS\ (T, H) = \frac{Len\ \left(MaxComSub\ (T, H\ )\right)}{Len\ (H)} \qquad (5.17)$$

5.1.3.3 Named Entity Matching

Using Named-Entity (NE) as a feature is helpful to improve the entailment recognition between any two sentences [23]. However, recognizing named entities is harder in Arabic than other languages due to the lack of capitalization and other challenges. Therefore, very few freely available tools are available for Arabic named entity recognition (NER) **[**103**].** The recognition of named entities in our work is performed using FARASA (QCRI) Arabic Language Technologies Tools & Demos **[105]**. Each Text-

Hypothesis pair is searched to detect named entities. Named entities appeared in Text and Hypothesis are compared. In case if there are entities in the support text match the entities in the corresponding hypothesis, we calculate the named entity feature as the ratio of the total number of matched named entities in the both the text and the hypothesis to the number of named entities in the hypothesis. The NEM feature is computed as follows:

$$\text{NEM}\ (T, H) = \frac{NE\ \ (T, H\ )}{NE\ \ (H)} \qquad (5.18)$$

5.1.3.4 Cosine Similarity

Cosine similarity is one of the measures of similarity that widely used in data mining and information retrieval to find the similarity between two documents or sentences [52]. It is a vector based similarity measure that measures the similarity between two n-dimensional vectors by computing the cosine of the angle between these vectors. The lower the angle between the two vectors is the more similar the two vectors are. In this research, we calculate the cosine similarity between each hypothesis (H) and support text (T) to measure how similar they are. Given the hypothesis vector H and the text vector T, the cosine similarity between the T and H is calculated using the dot product and magnitude as follows:

$$Cosine\ \ (T, H) = \frac{\sum_{i=1}^{n} (t_i \times h_i)}{\sqrt{\sum_{i=0}^{n} {t_i}^2 \ \times \ \sum_{i=0}^{n} {h_i}^2}} \qquad (5.19)$$

Where, t_i are the elements within the vector of a text and h_i are the elements within the vector of the hypothesis.

5.1.3.5 POS Similarity

The parts of speech and named entities can give useful indication for entailment recognition if they are given more attention [**70**]. Part Of Speech (POS) tagger is natural language processing tool that used to assign a syntactic role for each word in a sentence depending on the way the word is being used. The freely available Arabic POS tagger MADA+TOKAN [59] is used to classify words into their part-of-speech in both the text (T) and the hypothesis (H). We only consider nouns, verbs. For all the considered parts of speech detected by the POS tagger, each text (T) and corresponding hypothesis (H) pair is checked to identify the commonly shared nouns, verbs and adjectives appeared in both of them [115].

A) Noun Matching

Each hypothesis (H) and corresponding text (T) are checked to identify the noun words that common between them. Then the feature is calculated as the ratio of the count of shared nouns between the text (T) and the hypothesis (H) to the count of nouns in the hypothesis (H).

$$POS\_N(T,H) = \frac{number\ of\ matched\ nouns\ between\ H\ and\ T}{total\ number\ of\ nouns\ in\ H} \qquad (5.20)$$

B) Verb Matching

Each H-T pair is checked to identify the matched verb words between them. The feature is computed as the ratio of the number of common verbs between the hypothesis (H) and the supporting text (T) to the number of verbs in the hypothesis (H).

$$POS\_V(T,H) = \frac{number\ of\ shared\ verbs\ between\ H\ and\ T}{total\ number\ of\ verbs\ in\ H} \qquad (5.21)$$

Chapter Summary

In this chapter, we talked about textual entailment recognition problem and how it can be considered as classification problem and solved by machine learning. Thereafter, we provided a detailed discussion about various types of features including lexical, semantic and syntactic features that we used to solve the problem of recognizing the entailment between the text and the hypothesis. Next chapter describes our approach to address the challenge of answer selection in Arabic QA system.

Chapter 6

System Description

In this chapter, we present our approach to address the challenge of answer selection in Arabic QA system based on Textual Entailment (TE) recognition. The approach consists of combining three feature sets to evaluate whether one of the candidate answers can be inferred from the text returned by the system. Our system is designed to utilize information on the lexical, syntactic and semantic level in order to recognize the entailment between the generated hypotheses (H) and the text (T). The core modules of the system are outlined and each module of these modules consists of number of submodules are also described in more details.

6.1 System Architecture

The core components of the system are three modules, Text Processing module, Question and Answer Processing module and Textual Entailment Recognition module. Each component of these modules consists of number of submodules in order to fulfill its task. Each of those modules will be described in the following sections. Our system architecture is inspired by the architecture of the best performing English systems in QA4MRE campaign [98]. The architecture of our system is presented in Figure 6.1.

6.1.1 Text Processing Module

Different kinds of preprocessing are performed over the input text as preparation step including tokenization, stop words removing, stemming and normalization.

6.1.1.1Tokenization

In order to be further processed by next modules, the text is segmented into sentences and each sentence is split into individual words (tokens). Tokenization is usually based on the spaces between words or stop marks.

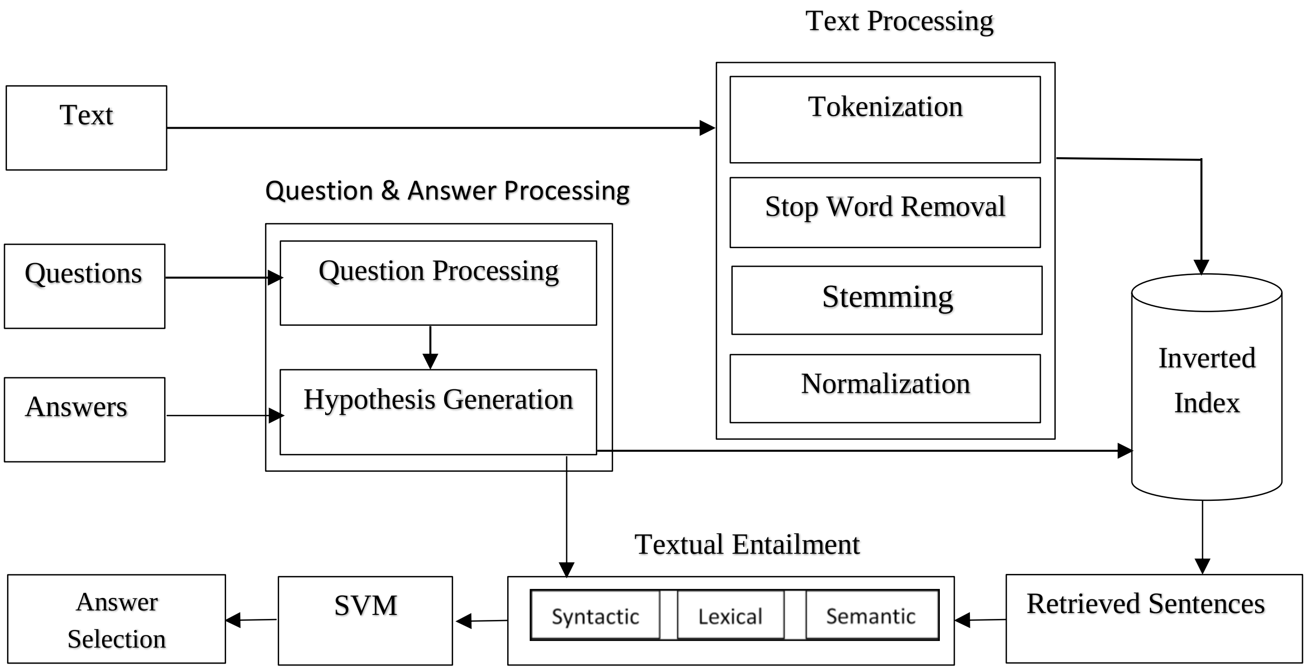


Figure 6.1: The proposed system architecture

For example, the sentence:

“ التغير المناخي الناتج من الاحتباس الحراري له تأثير سلبي على نظم المياه العذبة حول العالم “

(*Climate change resulting from global warming has a negative impact on freshwater systems around the world*)

After the process of tokenization is converted to the following list of tokens: (العالم, حول, العذبة, المياه,نظم, على, سلبي, تأثير, له, الحراري, الاحتباس, من, الناتج, المناخي,التغير).

6.1.1.2 Stop-words Removal

In order to improve the performance of matching process and produce more accurate results in the sentences retrieval step, the stop words are removed. Stop words are words that do not carry sense by themselves and rarely add any value to a search. These include but are not limited to the following: (Subjective pronouns: اسماء الاشارة) such as: " هو " (hw: he), for singular masculine " هي " (hy: she) for singular feminine, " هما " (hmA: they) for dual masculine and feminine , " هم " (hm: they) for plural masculine and " هن " (hn: they) for plural feminine. Relative nouns: الاسماء الموصولة) such as: "الذى " (Al*y: who) for singular masculine "التى " (Alty: who) for singular feminine, "اللتان " (AlltAn: who) for dual feminine , "اللذان " (All*An: who) for dual masculine ," الذين " (Al*yn: who) for plural masculine and "اللواتى/اللاتى/اللائى " (AllwAty/AllAty/AllA}y: who) for plural feminine.

6.1.1.3 Stemming

Stemming is a process for reducing each word in the sentence to its stem. A stem of a word is the part left after the affixes (prefixes, infixes and suffixes) have been removed. In our module, the text words and hypothesis words are converted to their stems. As an example, the stem of the plural word "كتب" (ktb: books) is the singular noun "كتاب" (ktAb: book), which is formed by deleting the infix alf (ا). Performing stemming increases the retrieval effectiveness and improves the performance of matching process.

6.1.1.4 Normalization

In Arabic, there are some letter forms often used in place of each other due to varying orthographic conventions and common spelling and typing mistakes. In order to enhance the research capabilities, our system performs normalization by removing special marks on letters and transforming some letters into a standard forms. These include:

- Replacing “ى” (ya) with “ي” (Yeh) (alefmaqsoura).
- Replacing “ ه ” (ha) with “ ة ” (ta marbouta)
- Replacing “ آ ”((alefmaad), “ أ ” (alef with hamza above), and “ إ ” (alef with hamza below) with “ ا ”(alef).

6.1.2 Question and Answer Processing Module

This module consists of two submodules: Question Processing and Hypothesis Generation.

6.1.2.1 Question Processing

Interrogative Particles (IP) أدوات الأستفهام are removed from each question. Interrogative Particles are the words that usually come at the beginning of the question. Such as: ” متى” (mtY:When),” اين “ (Ayn: Where), ” من “ (mn: Who), ”كم ” (km: How) ” ما“ (mA: What), and ” أى “(Ay: Which). An answer template is added to each question to be replaced by each option of the candidate answers in the next step. For example, the question:

ما هو السبب الاقتصادي للانعدام التام تقريبا من الحصول على أدوية مضادات الفيروسات الرجعية للمرضى في أفريقيا؟

What is the economic reason for the almost total lack of access to ARV drugs for patients in Africa?

is converted to:

السبب الاقتصادي للانعدام التام تقريبا من الحصول على أدوية مضادات الفيروسات الرجعية للمرضى في أفريقيا </answer>.

The economic reason for the almost total lack of access to ARV drugs for patients in Africa ***</answer>.***

6.1.2.2 Hypothesis Generation

Now for each answer option, for a given question, a corresponding hypothesis (H) is built by replacing each answer template with each answer choice to form a hypothesis. For the above example, the following hypotheses are generated for each of the candidate answers:

H_1:

السبب الاقتصادي للانعدام التام تقريبا من الحصول على أدوية مضادات الفيروسات الرجعية للمرضى في
أفريقيا < توافر العقاقير المضادة للفيروس البلدان الغنية >

The economic reason for the almost total lack of access to ARV drugs for patients in Africa <the availability of ARVs in wealthy countries>

H_2:

السبب الاقتصادي للانعدام التام تقريبا من الحصول على أدوية مضادات الفيروسات الرجعية للمرضى في
أفريقيا < ارتفاع أسعار الأدوية>

The economic reason for the almost total lack of access to ARV drugs for patients in Africa < the high drug prices>

H_3:

السبب الاقتصادي للانعدام التام تقريبا من الحصول على أدوية مضادات الفيروسات الرجعية للمرضى في أفريقيا < إلغاء الديون الخارجية للحكومات الأفريقية>

The economic reason for the almost total lack of access to ARV drugs for patients in Africa < the external debt cancellation for the African governments>

H_4:

السبب الاقتصادي للانعدام التام تقريبا من الحصول على أدوية مضادات الفيروسات الرجعية للمرضى في أفريقيا < أرباح شركات الأدوية>

The economic reason for the almost total lack of access to ARV drugs for patients in Africa < the profits of pharmaceutical companies>

H_5:

السبب الاقتصادي للانعدام التام تقريبا من الحصول على أدوية مضادات الفيروسات الرجعية للمرضى في أفريقيا < عدم وجود خطط سياسية>

The economic reason for the almost total lack of access to ARV drugs for patients in Africa < the lack of political plans>

6.1.2.3 Keywords Identification

After hypotheses generation, each hypothesis is stemmed using Arabic ISRI (Information Science Research Institute) root stemmer and the stop words are removed to identify the keywords. Then the hypothesis keywords are used to search the inverted index to retrieve the most relevant sentences. For the same above example, after we remove the stop words and stem the words, we got the following results:

H_1:

سبب اقتصاد انعدام تام تقريب حصول ادو مضادا فيروس رجع مرضي افريقي توافر عقاقير مضاد بلد الغن

H_2:

سبب اقتصاد انعدام تام تقريب حصول ادو مضادا فيروس رجع مرضي افريقي ارتفاع اسعار ادو

H_3:

سبب اقتصاد انعدام تام تقريب حصول ادو مضادا فيروس رجع مرضي افريقي الغاء الد خارج حكوم افريق

H_4:

سبب اقتصاد انعدام تام تقريب حصول ادو مضادا فيروس رجع مرضي افريقي ارباح شركا ادو

H_5:

سبب اقتصاد انعدام تام تقريب حصول ادو مضادا فيروس رجع مرضي افريقي عدم جود خطط سياس

6.1.2.4 Sentences Retrieval

After searching the hypothesis keywords against the inverted index, a set of sentences are retrieved for each query. Each sentence from the top retrieved sentences is defined as a Text (T) to be used for further processing with the associated Hypothesis (H).

6.1.3 Textual Entailment Recognition module

In order to recognize the entailment, for each given question, each defined Text (T) is paired with the corresponding generated Hypothesis (H). We considered seventeen features (features have been described in details in Chapter five.) to detect the entailment based on semantic, syntactic and lexical information.

6.1.3.1 Semantic Entailment

The semantic entailment submodule is based on two types of features, Semantic Similarity features and Lexical Semantic features.

6.1.3.1.1 Semantic Similarity Entailment

Arabic WordNet (AWN) is utilized to calculate the semantic similarity between each text (T) and hypothesis (H). We use three different Path-based measures to calculate the similarity between the T- H pairs.

A) WuP

We use WuP measure [121] to calculate the similarity between each word in the hypothesis (H) with all words in the text (T) using Arabic WordNet. The calculation is based on the depth of the two senses in the taxonomy and the distance of their LCS. The returned score denotes how these senses are similar. The measure is defined by equation 5.11 in Chapter 5, Section 5.1.2.1.

B) Path

Each word in the Text-Hypothesis pair is checked to see if it belongs to AWN. If we find them, we determine the similarity between the two words according to the Path measure using equation 5.12 in Chapter 5, Section 5.1.2.2.

C) LCH

We utilize the measure Leacock & Chodorow (LCH) [80] to compute the similarity between each Text-Hypothesis pair word senses through AWN. The similarity calculated based on equation 5.13 in Chapter 5, Section 5.1.2.2.

6.1.3.1.2 Lexical Semantic Entailment

This submodule uses the Arabic WordNet (AWN) [31] relations to perform a semantic matching between the Text and the Hypothesis. The approach is based on three types of matching: synonymy, hyponymy, hypernym.

A) Synonymy Matching

For each hypothesis and corresponding text, all the nouns, verbs and adjectives that are not stop words are compared to find synonyms using AWN. Synonymy matching feature calculates the overlap between the synonym words in the hypothesis (H) that match in the corresponding text based on AWN. The measure is defined by equation 5.9 in Chapter 5, Section 5.1.2.1.

B) Hyponym and Hypernym Matching

Nouns and verbs in the Text (T) and the Hypothesis (H) pair are checked if they are hyponyms or hypernyms. Hyponymy and hypernym matching feature is based on the overlap between number of hypothesis words that are hyponyms or hypernyms of other words in the text and the total number of the hypothesis words. The measure is calculated according to Equation 5.10 in Chapter 5, Section 5.1.2.1.

6.1.3.2 Lexical Entailment

The lexical entailment submodule is based on seven features: Unigram Match, Bigram Match, Trigram Match, Longest Common Subsequence (LCS), Named Entity Matching, Cosine Similarity and POS feature. These features have been described in more details in Section 5.1.3 of Chapter five.

6.1.3.2.1 Unigram Match

Each Text-Hypothesis pair is checked to calculate the number of the similar words appeared in both of them. This process calculates the ratio of the count of shared unigrams between each text (T) and the corresponding hypothesis (H) to the count of unigrams in hypothesis (H). The feature is computed based Equation 5.14 in Chapter 5, Section 5.1.3.1.

6.1.3.2.2 Bigram Match

Each Text-Hypothesis pair is checked to count the number of shared bigrams between the text (T) and the associated hypothesis (H). This process is computed based on Equation 5.15 in Chapter 5, Section 5.1.3.1.

6.1.3.2.3 Trigram Match

Each text (T) and support hypothesis (H) pair is checked to calculate the number of the trigrams words appeared in both of them. The feature is computed according to Equation 5.16 in Chapter 5, Section 5.1.3.1.

6.1.3.2.4 Longest Common Subsequence (LCS)

The system measures the longest common subsequence between the text (T) and the hypothesis (H) by searching the common subsequence with maximum length. The LCS feature is described by Equation 5.17 in Chapter 5, Section 5.1.3.2.

6.1.3.2.5 Named Entity Matching

Each Text-Hypothesis pair is searched to detect named entities. Named entities appeared in Text and Hypothesis is compared. In case if there are entities in the support text match the entities in the corresponding hypothesis, the system calculates the named entity feature as the ratio of the total number of matched named entities in the both the text and the hypothesis to the number of named entities in the hypothesis. The Named Entity matching feature is defined by Equation 5.18 in Chapter 5, Section 5.1.3.3.

6.1.3.2.6 Cosine Similarity

Given the hypothesis vector H and the text vector T, the system calculates the cosine similarity between the text (T) and the hypothesis (H) vectors based on Equation 5.19 in Chapter 5, Section 5.1.3.4.

6.1.3.2.7 POS Similarity

Each text (T) and corresponding hypothesis (H) pair is checked to identify the common shared POS appeared in both of them. Then the feature is calculated as the ratio of the number of shared POS between the text (T) and the corresponding hypothesis (H) to the number of POS in the hypothesis (H). The features are defined by Equations 5.20 and 5.21 in Chapter 5, Section 5.1.3.5.

6.1.3.3 Syntactic Entailment

Four syntactic features are used to compare the dependency relations between the text (T) and the hypothesis (H). The features are Subject – Verb, Object – Verb, Subject – Subject, and Object – Object. The features have been described in details in Section 5.1.1.2 of Chapter five.

6.1.3.3.1 Subject-Verb Matching (SBJ)

Subjects and verbs in the H are compared with subjects and verbs in the corresponding T. If both the T and the H have the same subject and verb words with SBJ relation, then a score is calculated based on the Equation 5.5 in Chapter 5, Section 5.1.1.3.

6.1.3.3.2 Object-Verb Matching (OBJ)

Objects and verbs in the H are compared with objects and verbs in the corresponding T. If both the T and the H have the same object and verb words with the OBJ relation, then a score is calculated as defined in Equation 5.6 in Chapter 5, Section 5.1.1.3.

6.1.3.3.3 Subject-Subject Matching

Each hypothesis (H) and corresponding text (T) pair is checked to identify common shared subjects presented in both of them. The feature is calculated as in Equation 5.7 in Chapter 5, Section 5.1.1.3.

6.1.3.3.4 Object-Object Matching

Each hypothesis (H) and corresponding text (T) pair is checked to identify the common objects between them. The feature is computed as given in Equation 5.8 in Chapter 5, Section 5.1.1.3.

6.1.4 Answer Selection

After TE module classifies the T-H pairs based on the trained model, the entailment decision is checked. The negative (non-entailed) pairs are ignored and the remaining pairs are ordered according to the scores obtained from the classifier. As the ultimate goal of our system is to select the correct answer and only one answer, for each question, the corresponding answer option to the hypothesis that receives the highest score is selected as the right answer.

Chapter Summary

In this chapter, we have introduced our approach to address the challenge of answer selection in Arabic QA system. The system is designed to utilize information on the lexical, syntactic and semantic level in order to recognize the entailment between the generated hypotheses (H) and the text (T). We presented the system architecture and the core modules of the system were outlined. Each module of these modules consists of number of submodules are also described in details. In the next chapter, we discuss the experiments and the results of applying our approach of answer selection in Question Answering system based on textual entailment recognition.

Chapter 7

Experimental Results and Evaluation

This chapter presents the experiments and the results of applying our approach of answer selection in Question Answering system based on textual entailment recognition. We start with a detailed description of the experimental settings, the datasets and the measures are used to evaluate the textual entailment recognition module and the overall system performance in answer selection task. Thereafter, the conducted experiments are explained in detail and the results are reported and analysed.

7.1 Experimental Setup

The evaluations of our system were carried out using two different datasets depending on the task being evaluated. The first dataset ArbTEDS[1] was used for training and testing the TE module while the second one QA4MRE[2] Arabic dataset is used to evaluate the overall system performance in answer selection task.

7.1.1 TE Dataset

Experimenting with textual entailment recognition requires datasets containing both positive and negative input T-H pairs. Unfortunately, there are fewer resources for TE for Arabic than for other languages, and to the best of our knowledge, the only TE dataset available for Arabic is

[1] http://www.cs.man.ac.uk/~ramsay/ArabicTE/

[2] http://nlp.uned.es/clef-qa/repository/qa4mre.php

ArbTEDS. It was built by Alabbas and Ramsay [17]. The dataset contains 600 pairs of Text-Hypothesis. These pairs were randomly selected from thousands of pairs collected from various subjects, such as sport, politics, business and general news. These T-H pairs were gathered by a variant of the headline:lead article technique that was used for building the first few RTE task datasets [41]. The dataset was built automatically by writing queries to Google and extracting text expressions that entail or do not entail the query. Eight Arabic native speaker (experts and non-experts) volunteer annotators were employed to annotate the different pairs as entailed or not entailed pairs manually using an online annotation system [15]. Figure 7.1 shows one of ArbTEDS text-hypothesis pair that was annotated as non-entailed pair.

***Text*:**

الرئيس الأمريكي باراك اوباما يزور بولندا فى المرحلة الاخيرة من
جولته الاوروبية ويلتقى فيها بزعماء وسط وشرقى اوروبا الاعضاء فى
الاتحاد الاوروبى لتعزيز العلاقات

```
Alr}ys Al>mryky bArAk >wbAmA yzwr bwlndA fy AlmrHlp Al>xyrp
mn jwlt h Al>wrwbyp w yltqy fy hA b zEmA^ wsT w $rqy >wrwbA
Al>EDA^ fy AlAtHAd Al>wrwby
```

“US President Barack Obama visits Poland in the last phase of his European trip and he will join leaders of Central and Eastern Europe nations that are members of the European Union for fence-mending”

Hypothesis:

.الرئيس اوباما يزور بولندا لتعزيز العلاقات بين البلدين

```
>wbAmA yzwr bwlndA l tEzyz AlElAqAt byn Albldyn
```

“President Obama visits Poland for fence-mending”

Judgement: NotEntails

Figure 7.1: Example of non-entailed text-hypothesis pair [15].

Each pair was marked up by three annotators who agreed on its entailment status. The pairs are marked as "Entails" if all three people who annotated it agreed that T entailed H and "NotEntails" otherwise. The corpus is balanced, with 300 Entails pairs and 300 NotEntails. Inter-annotator agreement was 74% for cases where all annotators agreed [17]. Table 7.1 shows that the average rates on the cases where the three annotators agree with at least one another annotator was 68%, which was less than those in the case only three annotators agree which was 80% [15].

Table 7.1: ArbTEDS annotation rates [15]

Agreement	YES	NO
=3 agree	478 (80%)	122 (20%)
>3 agree	409 (68%)	191 (32%)

Statistical analysis of the dataset suggests that sentences length could be one of the reasons behind that because people usually find long sentences harder to understand than short ones, and as a result, they disagree about the un-comprehended sentences more than about comprehended ones. Table 7.2 summarises the average annotation rates according to the text's length [15].

Table 7.2: Sentences' range annotation rates [15]

T's length	#pairs	#YES	At least one disagree
<20	131	97	34
20-29	346	233	113
30-39	110	69	41
>39	13	10	3
Total	600	409	191

The sentences were parsed using a combination of MADA [59] for tagging and MSTParser [86] for parsing. This combination obtained 81% labelled accuracy when tested on the Penn Arabic TreeBank. The parses are recorded in CoNLL format [17]. Pairs are presented as first simple text strings in Buckwalter transliteration [33], followed by a judgement (Entail/NotEntails), followed by the CoNNL format analysis of each sentence. Figure 7.2 presents the text-hypothesis pair shown in Figure 7.1 in CoNNL format [17].

```
Text:
Premise (Parsed):
1       Alr}ys  Alr}ys  DET+NOUN DET+NOUN -       5
2       Al>mryky Al>mryky DET+ADJ DET+ADJ -       1
3       bArAk   bArAk   NOUN_PROP       NOUN_PROP       -
        1
4       >wbAmA  >wbAmA  NOUN_PROP       NOUN_PROP       -
        3
5       yzwr    yzwr    IV      IV      -       0       ROOT
        -
6       bwlndA  bwlndA  NOUN_PROP       NOUN_PROP       -
        5
7       fy      fy      PREP    PREP    -       5       DEP
        -
8       AlmrHlp AlmrHlp DET+NOUN DET+NOUN -       7
9       Al>xyrp Al>xyrp DET+ADJ DET+ADJ -       8       DEP
        -
10      mn      mn      PREP    PREP    -       8       DEP
        -
11      jwlt    jwlt    NOUN    NOUN    -       10      OBJ
        -
12      h       h       POSS_PRON       POSS_PRON       -
        11
13      Al>wrwbyp       Al>wrwbyp       DET+ADJ DET+ADJ -
        11
14      w       w       CONJ    CONJ    -       5       COORD
        -
15      yltqy   yltqy   IV      IV      -       14      DEP
        -
16      fy      fy      PREP    PREP    -       15      DEP
        -
17      hA      hA      PRON    PRON    -       16      OBJ
        -                       b       PREP    PREP    -
        15      DEP     -       -
19      zEmA^   zEmA^   NOUN    NOUN    -       20      DEP
        -
20      wsT     wsT     NOUN    NOUN    -       18      OBJ
        -
21      w       w       CONJ    CONJ    -       20      COORD
        -
```

```
22       $rqy     $rqy     NOUN     NOUN     -        21       DEP
         -
23       >wrwbA   >wrwbA   NOUN_PROP         NOUN_PROP         -
         22
24       Al>EDA^  Al>EDA^  DET+NOUN DET+NOUN -        20
25       fy       fy       PREP     PREP     -        15       DEP
         -
26       AlAtHAd  AlAtHAd  DET+NOUN DET+NOUN -        25
27       Al>wrwby Al>wrwby DET+ADJ  DET+ADJ  -        26

Hypothesis
1        >wbAmA   >wbAmA   NOUN_PROP         NOUN_PROP         -
         2
2        yzwr     yzwr     IV       IV       -        0        ROOT
         -
3        bwlndA   bwlndA   NOUN_PROP         NOUN_PROP         -
         2
4        l        l        PREP     PREP     -        2        DEP
         -
5        tEzyz    tEzyz    NOUN     NOUN     -        6        DEP
         -
6        AlElAqAt AlElAqAt DET+NOUN DET+NOUN
7        byn      byn      NOUN     NOUN     -        8        DEP
         -
8        Albldyn  Albldyn  DET+NOUN DET+NOUN -        2
```

Figure 7.2: Example of text-hypothesis pair in CoNNL format [17].

7.1.2 QA4MRE dataset

In order to measure the performance and the effectiveness of our approach, the experiments have been conducted using QA4MRE (Question Answering for Machine Reading Evaluation) task dataset. QA4MRE is a task that was introduced for the multilingual QA track of CLEF for the first time in 2011. It was designed to skip the answer generation and give more attention to answer selection and validation subtasks over the IR based tasks of passage retrieval. QA4MRE is used interchangeably with task of Answer Selection and Validation [30]. In 2012 reading tests and questions were made available in seven languages, namely: Arabic, Bulgarian, English, German, Italian, Romanian, and Spanish. The task consisted of four topics: Music and Society, Climate Change, AIDS and Alzheimer. Each topic had four reading tests. Each reading test provided with one single document followed by 10 questions and a set of five choices per question. Table 7.3 presents the distribution of question types in QA4MRE@CLEF 2012 [99]. The total set included 16 test documents, 160 questions and 800 choices. There is one and only one correct option for each question and the role of the system is to select the most appropriate answer option. A sample of QA4MRE 2012 Arabic dataset is shown in Figure 7.3. The dataset is defined using the XML tags as follows:

- "t_id": is the topic id.
- "t_name": is the topic title.
- "r_id": is the unique id of the reading test.
- "doc": is the test document against which the questions are being asked.
- "d_id": is the id of the test document.
- "q_id": is the question id.
- "q_str": is the question (UTF-8 encoded) string.
- "a_id": is the answer id.

Table 7.3: Distribution of question types in QA4MRE [99].

Question type	Example	Number of questions	Percentage (%)
Causal	What was the cause of event X?	36	22.50
Factoid	where, when, who	36	22.50
Purpose	what was the reason for doing X?	27	16.88
Method	How did X do Y?	30	18.75
Which is true	What can a 14 year old girl do?	31	19.38
Total		160	100

<?xml version="1.0" encoding="UTF-8"?>

<test-set>

- <topic t_name="**AIDS**" t_id="**1**">

  - <reading-test r_id="**1**">

تحدي نساء أفريقيات لسياسة بوش تجاه الإيدزتشع الهيبة من ريبيكا لولوسولي بينما تغطي جبهتها ورقبتها وصدرها <doc d_id="**1**"> ومعصميها طبقات من الزخارف الخرزية، تواجه باسمة جمهور من المستمعين من طلبة الجامعات الأمريكية، مع أن الموضوع بالنسبة لهم هو كناية عن البؤس والضعف. تتحدث ريبيكا عن الإيدز في أفريقيا، خاصة في قرية أموجا بكينيا التي يسكنها قبيلة سامبورو من سكان كينيا الأصليين. "لسنوات يموت الناس ولا أحد يعلم لماذا،" تذكر ريبيكا. "الآن نعرف أنه يمكننا تجنب الإيدز، ولكن فقط عبر تغييرات كبيرة في حياتنا."بفضل النشطاء في مجال الصحة العامة والعدل الاجتماعي في أفريقيا مثل ريبيكا، يزداد عدد الذين يعرفون أن أفريقيا جنوب الصحراء هي الآن المركز الأساسي لوباء الإيدز: فأفريقيا موطن 3 من كل 4 في العالم يموتون بسبب المرض، وهي القارة التي بها ثلثي حاملي الفيروس في العالم (أكثر من 25 مليون نسمة). ولكن لا يعرف الجميع أن معظم هؤلاء المرضى هم من النساء، وأن حديثي السن منهن يزيد معدل إصابتهن بمعدل من 3 إلى 4 أضعاف الرجال من نفس الشريحة السنية. فيجب أخذ التالي في الاعتبار:• منذ الثمانينات عندما بدأ ظهور الإيدز، طالبت الولايات المتحدة بسياسات اقتصادية قاسية في الدول الفقيرة. ففي أفريقيا، قطعت هذه السياسات ميزانيات الصحة إلى النصف بينما كانت أنظمة الصحة العامة تحتاج هذا الدعم لمواجهة الإيدز. واليوم، فإن هذا الوباء هو العائق الأكبر الوحيد تجاه التنمية الاقتصادية في أفريقيا.• ولزيادة الأرباح الفاحشة لشركات الأدوية الأمريكية، فإن إدارة بوش منعت بيع الأدوية الرخيصة الثمن التي كان يمكنها هذا الصندوق، والمقررة ب3.5 مليار دولار، أو حوالي ثلث الصندوق، فإن الولايات المتحدة وعدت فقط بإنفاق 0.6 مليار دولار لعامي 2006-2007.ولا يعتبر عرض اسقاط الديون المقدم من مدموعة الدول الثماني (أغنى دول العالم) الشيء الكثير بالنسبة للمتعايشين مع الإيدز في أفريقيا. يعتقد الكثير أن ذلك سيوفر الأموال لمحاربة الإيدز، ولكن ليس هناك آلية محددة لتحقيق ذلك على الأرض. بل إن هذا العرض يجعل معظم الدول الأفريقية تنفق على خدمة الديون 4 أضعاف ما تنفقه على الصحة والتعليم - أهم قطاعين للقضاء على الإيدز. إذا تم تحويل خدمة الديون إلى جهود محاربة الإيدز فسيوفر ذلك 15 مليار دولار في العام - وهذا بالضبط ما تحتاجه الأمم المتحدة لتمويل برامجها ضد الإيدز. نعلم أن بإمكان البنك الدولي وصندوق النقد الدولي إلغاء كامل ديول الدول الفقيرة بدون الإخلال ببرامجهما، ولكن أكبر مساهم في هاتين المؤسستين، الولايات المتحدة، تعارض إلغاء الديون الغير مشروط. ولا يتعلق هذا بالمال، وهو قدر ضئيل بالنسبة لاقتصاد الولايات المتحدة. بل يتعلق بإجبار الحكومات الأفريقية على تنفيذ سياسات تتفق مع مصالح الولايات المتحدة.أصدرت الأمم المتحدة الأسبوع الماضي تقريرها السنوي عن أزمة الإيدز العالمية، ويعبر عن أخبار سيئة في المعظم، ولكنه أيضاً أشار إلى برامج مواجهة ومعالجة قاسية لتقليل معدل الإصابة بالفيروس في كينيا من 10% إلى 7% بين التسعينات وعام 2003، مع تقليل معدلات نقل المرض من الأمهات الحوامل إلى أطفالهن في كينيا من 28% إلى 9% في نفس المدة.تأكدت ريبيكا لولوسولي ، وهي منظمة دولية لحماية حقوق MADRE بنفسها من أهمية الدمج بين برامج العلاج والمنع في مواجهة الإيدز. ففي العامين الماضيين، بدأت ريبيكا العمل مع الإنسان للمرأة، لجلب من يساهم في التدريب على منع المرض في مجتمعها. أعلى ما نصبو إليه هو منع الإصابة بالمرض من الأساس، ولذلك، فنحن النسوة يجب علينا أن يكون لنا القدرة على قول لا بدون التعرض للعنف أو الإجبار. يجب على النساء أن يكون لهن الحق في ملكية ووراثة الأرض التي تمكنهم من </doc> إطعام أنفسهن وأطفالهن. هذا هو طريق الوصول للصحة. "تغيير التقاليد ليس سهلاً"، تقول ريبيكا بابتسامة عريضة.

  - <q q_id="**1**">

<q_str>

ما هو السبب الاقتصادي للانعدام التام تقريبا من الحصول على أدوية مضادات الفيروسات الرجعية للمرضى في أفريقيا؟ </q_str>

<answer a_id="**1**">توافر العقاقير المضادة للفيروس في البلدان الغنية</answer>

<answer a_id="**2**" correct="**Yes**">ارتفاع أسعار الأدوية</answer>

<answer a_id="**3**">إلغاء الديون الخارجية للحكومات الأفريقية</answer>

<answer a_id="**4**">أرباح شركات الأدوية</answer>

<answer a_id="**5**">عدم وجود خطط سياسية</answer>

</q>

Figure 7.3: Sample of QA4MRE Arabic data-set as XML format [99]

7.2.1 TE Evaluation Measures

There are three widely known evaluation metrics used in textual entailment recognition in order to measure the performances of different approaches. Namely: recall, precision, F-measure. These measures are defined as follows:

$$\text{Precision: P} = \frac{tp}{tp+fp} \qquad (7.1)$$

$$\text{Recall: R} = \frac{tp}{tp+fn} \qquad (7.2)$$

where:

tp: True positives are the numbers of pairs that have correctly been classified as positive textual entailment pairs.

fp: False positives are the numbers of pairs that have incorrectly been classified as positive textual entailment pairs.

tn: True negatives are the numbers of pairs that have correctly been classified as negative textual entailment pairs.

fn: False negatives are the numbers of pairs that have incorrectly been classified as negative textual entailment pairs.

$$F - measure: F = \frac{(1+\beta^2)PR}{(\beta^2 P)+R} \qquad (7.3)$$

$$F\beta = 1 = \frac{2PR}{P+R} \qquad (7.4)$$

where:

β is a parameter indicating importance of recall (R) and precision (P). The value of β controls the trade-off between recall and precision. When the importance of recall and precision are equal, β is assigned to 1[103].

7.2.2 Answer Selection Evaluation Measures

For our system to be comparable with other system's performance, we used the same metric used by the QA4MRE systems [100]. The measure is called C@1. It gives a partial credit for systems that leave some questions unanswered instead of answering them wrongly. C@1 is represented by the following formula:

$$\mathrm{C@1} = \frac{1}{n}\left(n_R + n_u \frac{n_R}{n}\right) \qquad (7.5)$$

where:

n_R: is the number of correctly answered questions

n: is the number of questions

n_u: is the number of unanswered questions

To measure the system performance considering only the number of correct answers, we used Accuracy measure. It is calculated by dividing the number of relevant items retrieved plus the number of not relevant items that are not retrieved by the number of all items [109].

$$\mathrm{Accuracy} = \frac{\mathrm{tp+tn}}{\mathrm{tp+fp+tn+fn}} \qquad (7.6)$$

Where:

tp: True positives

tn: True negatives

fp: False positives

fn: False negatives

7.3 Experiments Results

This section reports the outcomes of the experimental testing that we conducted to evaluate our methods. First, we discuss the results of applying TE over ArbTEDS dataset utilizing our selected features. Then, an ablation test to assess the contribution of our selected features and how they affect the behavior of our TE module is presented and the results are discussed in section 7.3.1.1. Second, the conducted experiments to evaluate the overall performance of the proposed system using QA4MRE dataset are described and discussed in detail. Thereafter, a comparison between the obtained results with other Arabic systems results to highlight the effectiveness of the chosen techniques is illustrated.

7.3.1 Textual Entailment results

The system is based on Support Vector Machine that utilizes information on the lexical, syntactic and semantic level in order to recognize the entailment between the generated hypotheses (H) and the retrieved text (T). Seventeen features have been extracted from the T-H pairs, and then the feature vectors are fed into the SVM. The classifier classifies the T-H pairs based on the trained model. For each pair (T, H), where H is a hypothesis and T is the corresponding text, we examine whether the TE module correctly predicts the class of their entailment “Yes” or “No”. The input of the Textual Entailment module is Text-Hypothesis pairs from ArbTEDS dataset and the output is these pairs with “Yes” or “No” annotations.

In our experiments, we have used the Arabic dataset we mentioned in section 7.1.1 for training and testing to evaluate the TE module. Given that the size of our used dataset is small and to make sure that our model is more generalizable, we divided the dataset into two sets where the size of the training set is tribble the size of the test set. 150 T-H pairs of the dataset were chosen in order to test the TE task performance and the remaining 450 pairs used to perform 10-fold Cross Validation in order to estimate the accuracy of our model when dealing with unseen data. We divided the dataset into 10 groups and for each group; we take the group as test data and the rest of the groups as training dataset. The average accuracy of each fold in our model reached a score of 79.70% which is promising compared to other Arabic systems using the same dataset. Now, we test our model on the 150 unseen data and evaluate the results in terms of recall and precision.

Among the 150 test examples, the entailment predictions made by our approach achieved a recall, precision and f-measure of 78.66%, 81.94% and 80.26% respectively, for "Yes" annotation. For "No" answers, the recall, precision and f-measure were 82.66%, 79.48% and 81.03% respectively. Our results for Textual Entailment recognition task are summarized in Table 7.4.

Table 7.4: Evaluation results for Textual Entailment recognition.

Entailment class	# of T-H pairs in the dataset	# of the entailments given by our module	# of the pairs that entailed correctly by our module	Recall	Precision	f-measure
Yes	75	72	59	78.66	81.94	80.26
No	75	78	62	82.66	79.48	81.03
Total/Average	150	150	121	80.66	80.71	80.64

7.3.1.1 Ablations Test and Results

An ablation test typically refers to removing one module at a time from a system, and then re-running the system with the other modules to see how that affects performance. Therefore, in order to evaluate the contribution of our individual feature sets on the dataset, we ran our systems in turn with each feature subset removed. Ablated features results are shown in Table 7.5.

Table 7.5: Ablation results on the ArbTEDS dataset

Features ablated		Entailment decision	Recall	Precision	f-measure
All Features		Yes	78.66	81.94	80.26
		No	82.66	79.48	81.03
Lexical Features	N-grams	Yes	61.22	59.12	60.15
		No	62.71	61.84	62.27
	Rest of Lexical	Yes	67.63	68.13	67.87
		No	66.97	67.34	67.15
Syntactic		Yes	73.38	74.73	74.10
		No	74.66	72.54	73.77
WordNet based features	Semantic Similarity	Yes	76.64	75.62	76.12
		No	76.52	74.66	75.57
	Lexical Semantic	Yes	77.62	78.64	78.12
		No	77.98	76.32	77.14

Table 7.5 shows the performance of the TE module on ArbTEDS dataset without different feature subsets each time. It is interesting to see that the most valuable subset among the features is the lexical features. It can be set alone a good baseline. When the N-gram features were excluded, the system produced scores of 61.22%, 59.12% and 60.15% for recall, precision and f-measure respectively in case of positive entailment decisions. For negative entailment decisions, the system reached scores of 62.71%, 61.84% and 62.27% for recall, precision and f-measure respectively. The second most useful features are: Longest Common Subsequence, Named Entity, Cosine similarity and POS features. By removing these features, the system

obtained about 67.63 % recall, 68.13% precision and 67.87% f-measure respectively for “Yes” entailment decisions, while for “No” entailment decisions, the results were 66.97% recall, 67.34% precision and 67.15% f-measure respectively. Syntactic features had less impact on the module performance with f-measure score of 74.10% for “Yes” entailment and f-measure score of 73.77% For “No” answers. On the other hand, it is surprising to see that the WordNet based features had very small effect on the system’s recall and precision. When the Semantic Similarity features were removed, the system’s recall and precision in both cases positive and negative entailment decisions have been slightly dropped reaching a scores of 76.12% and 75.57% of f-measure for “Yes” annotations and “No” annotations respectively. Whilst, when we look at the Lexical Semantic features ablation result, we notice that these features had almost no effect on the module performance. In general, despite its wide coverage, AWN has limited improvement on the TE module performance through the lexical semantic features and semantic similarity features subsets compared to lexical features and syntactic features. This is due to the following reasons: First, these features based on word to word similarity metrics and they calculate the similarity between words at the semantic level, without considering their corresponding contexts. Second, AWN is more appropriate for representing relations between concepts such as common nouns but less for verbs. This is probably due to the fact that the relations between events such as verbs are more complex and have more internal structure than nouns and that the AWN`s verb hierarchy is not as deep as that for nouns. Another reason is nouns and verbs are grouped in separate hypernym hierarchies in AWN, therefore calculating similarities between verbs and nouns is not available.

7.3.2 Answer Selection results

Since the objective of our system is to answer the input question by selecting one answer from the five alternative answers, we have carried out experiments in order to measure the quality of the proposed methods and evaluate the overall system performance. The experiments were carried out using QA4MRE dataset as described in Section 7.1.2, where the system was required to give only one answer for each question. As we mentioned in Section 7.2.2, for our system to be comparable with other system's performance, the evaluation of the conducted experiments was measured according to the same widely used metrics used by the QA4MRE track. Namely, C@1 which is represented by Equation (7.5) and Accuracy which is defined by Equation (7.6). It is worth mentioning here that our system used only the Arabic dataset and did not utilize the background collection. Table 7.6 gives statistics about the obtained results in terms of questions and the overall accuracy and C@1 performance of our system.

Table 7.6: Obtained results and the overall accuracy and C@1 performance.

Discerption	Numbers	%
Total # of questions	160	100
# of answered questions	123	76.87
# of unanswered questions	37	23.13
# of correctly answered questions	84	52.50
# of incorrectly answered questions	39	24.38
Over all Accuracy	52.5	
C@1 measure	64.64	

The overall performance of the proposed system reached an accuracy of 52.5%. The number of all answered questions represents more than 76% of the questions of QA4MRE dataset. The system answered 123 out of 160 questions. From those, 84 correct answers, 39 incorrect answers and 37 unanswered resulting an C@1 score of 64.64. Figure 7.4 shows the questions distribution according to their answers.

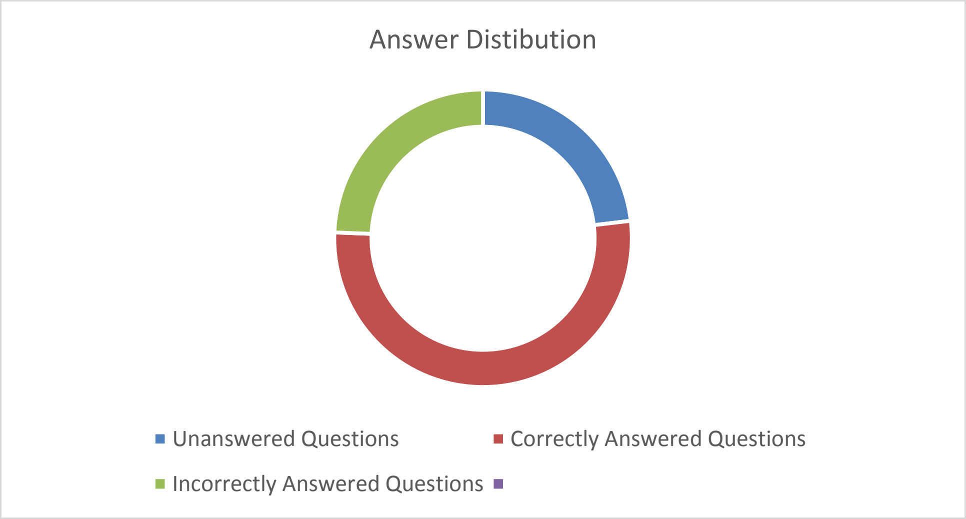


Figure 7.4: The questions distribution according to their answers

To gain a deeper understanding of our system performance and how it behaves with different type of questions, we evaluated the system with each question type individually. The number of questions that were correctly answered and those wrongly answered from each type of question as well as the overall and detailed accuracy are illustrated in Table 7.7.

Table 7.7: Obtained results per question type and detailed accuracy

Question type	Total # of questions	Percentage (%)	# of answered questions	# of unanswered questions	# of correctly answered questions	# of incorrectly answered questions	Accuracy
Factoid	36	22.50	33	3	29	4	80.55%
W-is-T	31	19.38	24	7	22	2	70.96%
Purpose	27	16.88	21	6	16	5	59.26%
Method	30	18.75	18	12	8	10	26.66%
Causal	36	22.50	27	9	9	18	25.00%
Total	160	100	123	37	84	39	52.50%

As we can notice from Table 7.7, the system performed the best on Factoid questions where over 80% of this type of questions was correctly answered. The second best score was achieved by the questions of type "Which is true" reaching an accuracy of 70.96%. The reason for our approach to be more effective in Factoid questions is that these questions are simple questions. They usually ask about entities such as location, person name or an organization and the answers are short sentences. These kinds of answers can be searched easily since they do not require external knowledge and/or complex inference. In contrast, the worst performance of our system was on both Causal and Method questions with an average accuracy of about 26.00%. These types of questions were expected to have lower results than other types since they tend to be more complex and ask about information that needs better understanding and requires more inference.

In regard to the system performance per topic, Table 7.8 lists the distributions of the answered and unanswered questions along with detailed and the overall C@1 results per topic.

Table 7.8: Detailed and overall C@1 evaluation measure per topic

Topic	# of questions	# of correctly answered questions	# of unanswered questions	C@1 measure
Climate Change	40	27	10	84.38
AIDS	40	25	13	82.81
Alzheimer	40	18	7	52.88
Music and Society	40	14	7	41.13
Total	160	84	37	64.64

As can be noticed from the Table 7.8, the C@1on each topic varies with "Climate Change" topic having the highest score at 84.38%, followed by topic "AIDS" with C@1 score of 82.81%. On the other hand, the system's C@1 score dropped when dealing with the other two topics' texts and questions "Alzheimer" and "Music and Society". Figure 7.5 displays a comparison between the four topics in terms of C@1 values.

In general, after further inspection we noticed that the system performance decreases when dealing with the following situations: First, domain specific concepts that need a specific background ontology. For example, when the system deals with a question such as: (ماهو النشاط البشري الذي يساهم في تغير المناخ؟ : *What is human activity that contributes to climate change*?), Arabic WordNet can succeed to expand the general words such as " نشاط "(n$AT : *activity*) to the word "مهنة" (mhnp : *profession*) as synonym and to the word " وَظِيفَة " (wZyfp : *job*) as hypernym which share the word " منصب" (mnSb: *position*) as synonym too. On the other hand, it could not expand domain specific word such as: “الطاعون الدموي” (AlTAEwn Aldmwy: *Blood Plague*). Second, questions are either complex questions that need domain the background collections to be answered or inferences questions that require composing several answers from different sentences for example, the question: (تشاجر نيويورك في مقره تلفزيونية خدمة مقدم أي بسبب والدها مع المؤلف؟: *Because of which television provider with headquarters in New York did the author quarrel with her father*?). Third, Questions with English acronyms, for example the question: (برنامج من الغرض هو ما GREET؟: *What is the purpose of the GREET software*?). The word “GREET” is not understandable and cannot be processed by Arabic tools. Fourth, questions do not begin with interrogative particles, for example the question: (الموسيقى عالم في تحكما عنصران هما ما عشر، التاسع القرن بحلول ؟: *By the 19th century, what are the two controlling elements of the music world*?. Fifth, questions and sentences with translation errors. For example, the question (الأمطار مياه هدر في يساهم الذي النشاط اسم : *Name an activity that contributes to waste rainwater* ?). Sixth, questions with negative terms such as (“ليس” (lys: not), “لن” (ln: won’t), …etc.). For example, the question: (أي من التالي ليس سببا للعدوى بفيروس نقص المناعة للنساء المتزوجات؟ : *Which of the following is not a cause of HIV infection for married women?*).

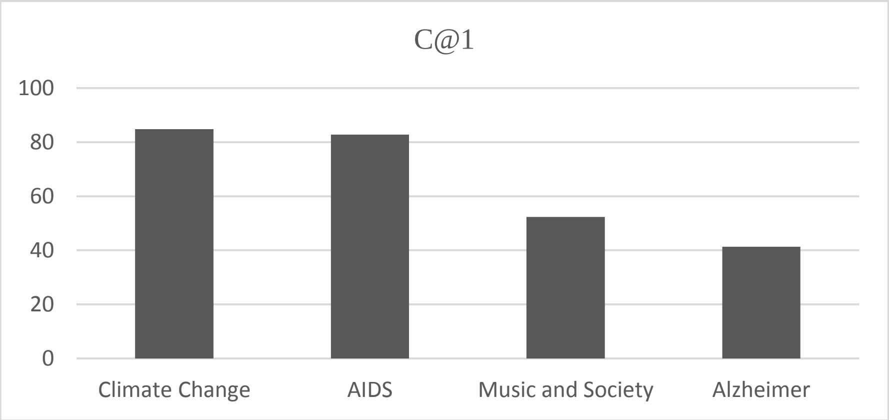


Figure 7.5: Overall C@1 evaluation measure per topic

7.3.2.1 Comparison with other systems

In order to highlight the effectiveness of the used approach, we compared the results achieved by our method over the QA4MRE dataset with those obtained by Arabic systems that participated in the same challenge. Figure 7.6 shows the comparison results.

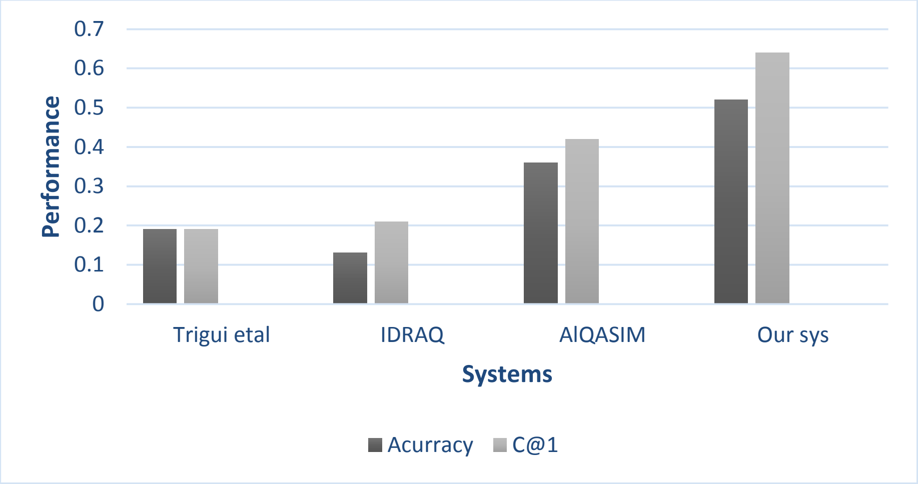


Figure 7.6: Performance comparison of our system with other systems

The comparison in Figure 7.6 shows clearly that our system performs significantly better than the Arabic systems that participated on QA4MRE main task in in terms of accuracy and C@1measures.

Chapter summary

In this chapter, we provided a detailed discussion about the experiments and the results of applying our approach of answer selection in Question Answering system based on textual entailment recognition. We first described the datasets and the measures were used to evaluate the TE recognition module and the overall system performance in answer selection task. Then, we detailed the conducted experiments, reported and analysed the obtained results. The obtained results show that our method helps significantly to tackle the problem of Answer Selection in Arabic Question Answering system. The size of dataset used in our experiments is relatively small. This might affected both learning of our model and evaluation of its performance. However, compared to other Arabic systems, the performance of our module has achieved fairly well and the results are encouraging.

Chapter 8

Conclusions and Future Work

8.1 Conclusion

In this dissertation, we have introduced a complete method to tackle the problem of Answer Selection in Arabic Question Answering system. The main objective of this work is to investigate the possibility of building an Answer Selection model for QA system that performs better than the state-of-the-art Arabic QA systems.

Answer selection is an important task for any QA system to perform. After the answer generation task generates a list of candidate answers to the input question, the answer selection component tries to select the best answer choice from the candidate answers suggested by the system. However, the selection process can be very challenging especially in Arabic due its particularities.

To address this challenge in Arabic, we proposed an approach to answer questions with multiple answer choices for Arabic. The approach based on Textual Entailment (TE) recognition method. The basic idea is to evaluate whether one of the candidate answers can be inferred from the text returned by the system. In case of a candidate answer is entailed by the supporting text, it then can be chosen as a correct answer.

Our work is the first work in Arabic Question Answering that combines three different sets of features that include lexical, semantic and syntactic features in one approach to solve the problem of textual entailment recognition in Arabic.

The developed approach employs Support Vector Machine that considers lexical, semantic and syntactic features in order to recognize the entailment between the generated hypotheses (H) and the text (T). Each sentence (T) is paired with the corresponding hypothesis (H) to represent T-H pair. Thereafter, features are extracted from the T-H pairs. These feature vectors are fed into the trained classifier in the TE module. The TE module classifies the T-H pairs based on the trained model.

In order to measure the performance and the effectiveness of the overall system, a set of experiments have been conducted using the Arabic dataset that provided by CLEF 2012 through the task of QA4MRE. The dataset was designed to skip the answer generation and give more attention to answer selection and validation subtasks over the IR based tasks. For performance evaluation of the TE module, the experiments were carried out using the Arabic dataset ArbTEDS that developed by Alabbas [15].

The evaluation results are satisfactory and encouraging considering the particularities of the Arabic and the high level of its complexity. The obtained results show that our method helps significantly to tackle the problem of Answer Selection in Arabic Question Answering system.

In order to highlight the effectiveness of the used techniques we have compared the results achieved by our method over the QA4MRE dataset with those obtained by Arabic systems that participated in the same challenge. The comparison showed clearly that our system performs significantly better than the Arabic systems that participated on QA4MRE main task in in terms of accuracy and C@1measures.

8.2 Future work

There is a plenty of room for more investigation to enhance the results of this work. During the development of this work, many issues, concerns and interesting questions have been raised. Therefore, our research will not be ended with the presentation of this dissertation. This work is the beginning of our study in the field. We are considering working on the following issues:

- The size of Arabic TE dataset used in our experiments is small. This might have affected both learning of our model and evaluation of its performance. Therefore, building a larger dataset will help us to evaluate how the system performance and its accuracy could be affected.

- Since complex questions need domain knowledgebase and background collections to be answered, we are planning to experiment with the background collection provided by QA4MRE task. External knowledge such as Wikipedia also can be used for further improvements.

- Anaphora is another problem needs to be addressed. It is a linguistic phenomenon of referring back to a previously mentioned item in the same text [69]. The process of resolving what a noun phrase, or a pronoun refers to is called *anaphora resolution*. Arabic text usually contains many anaphora expressions. Resolving the anaphora in our system will decrease the ambiguity of the sentences and improve the answer selection process through increasing the chance of matching between the text and the hypothesis.

- Negative terms can completely change the meaning of a text. In some cases, the text is saying the opposite of what the hypothesis is saying. Thus, dealing with this issue will increase the system performance to answer questions.

Bibliography

[1] Abdelbaki, H., M. Shaheen, and O. Badawy. "ARQA high-performance Arabic question answering system." In Proceedings of Arabic Language Technology International Conference (ALTIC). 2011.

[2] Abdelnasser, Heba, Reham Mohamed, MahaRagab, Alaa Mohamed, Bassant Farouk, Nagwa El-Makky, and Marwan Torki. "Al-Bayan: An Arabic Question Answering System for the Holy Quran." ANLP 2014 (2014): 57.

[3] Abouenour, Lahsen, and Karim Bouzoubaa. "Using an Arabic ontology to improve the Q/A task." 11th International Business Information Management Association Conference IBIMA, Cairo, Egypt; 01/2009.

[4] Abouenour, Lahsen, Karim Bouzouba, and Paolo Rosso. "An evaluated semantic query expansion and structure-based approach for enhancing Arabic question/answering." International Journal on Information and Communication Technologies 3, no. 3 (2010): 37-51.

[5] Abouenour, Lahsen, Karim Bouzoubaa, and Paolo Rosso. "IDRAAQ: New Arabic question answering system based on query expansion and passage retrieval." (2012).

[6] Abouenour, Lahsen, Karim Bouzoubaa, and Paolo Rosso. "Improving Q/A Using Arabic Wordnet." In Proc. The 2008 International Arab Conference on Information Technology (ACIT'2008), Tunisia, December. 2008.

[7] Abouenour, Lahsen, Karim Bouzoubaa, and Paolo Rosso. "Structure-based evaluation of an Arabic semantic Query Expansion using the JIRS Passage Retrieval system." In Proceedings of the EACL 2009 Workshop on Computational Approaches to Semitic Languages, pp. 62-68. Association for Computational Linguistics, 2009.

[8] Abouenour, Lahsen, Karim Bouzoubaa, and Paolo Rosso. "Three-level approach for passage retrieval in Arabic question/answering systems." In Proc. Of the 3rd International Conference on Arabic Language Processing CITALA2009, Rabat, Morocco. 2009

[9] Abdelali, Ahmed, Kareem Darwish, Nadir Durrani, and Hamdy Mubarak. "Farasa: A fast and furious segmenter for arabic." In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pp. 11-16. 2016.

[10] Agichtein, Eugene, David Carmel, Donna Harman, Dan Pelleg, and Yuval Pinter. "Overview of the trec 2015 liveqa track." In The Twenty-Fourth Text REtrieval Conference (TREC 2015) Proceedings. National Institute of Standards and Technology (NIST). 2015

[11] Agichtein, Eugene, Walt Askew, and Yandong Liu. "Combining Lexical, Syntactic, and Semantic Evidence for Textual Entailment Classification." In *TAC*. 2008.

[12] Ahmed, K. "The Arabic language: Challenges in the modern world." International Journal for Cross-Disciplinary Subjects in Education (IJCDSE) 1, no. 3 (2010): 283-292.

[13] Ahmed, Waheeb, Ajusha Ahmed, and Anto P. Babu. "Web-Based Arabic Question Answering System using Machine Learning Approach." *International Journal of Advanced Research in Computer Science* 8, no. 1 (2017).

[14] Akour M., Abufardeh S., Magel K. and Al-RadaidehQasem A., (2011),"QArabPro: A Rule BasedQuestion Answering System for Reading Comprehension Tests in Arabic", American Journal of Applied Sciences, 8(6): pp. 652-661.

[15] Alabbas, Maytham. "A Dataset for Arabic Textual Entailment." In *Proceedings of the Student Research Workshop associated with RANLP 2013*, pp. 7-13. 2013.

[16] Alabbas, Maytham. "ArbTE: Arabic textual entailment." In *Proceedings of the Second Student Research Workshop associated with RANLP 2011*, pp. 48-53. 2011.

[17] Alabbas, Maytham, and Allan Ramsay. "Natural language inference for Arabic using extended tree edit distance with subtrees." *Journal of Artificial Intelligence Research* 48 (2013): 1-22.

[18] Al Chalabi, Hani Maluf. "Question Processing for Arabic Question Answering System." (2015).

[19] Al Chalabi, Hani Maluf, Santosh Kumar Ray, and Khaled Shaalan. "Question classification for Arabic question answering systems." In *Information and Communication Technology Research (ICTRC), 2015 International Conference on*, pp. 310-313. IEEE, 2015.

[20] Aliyeva, Narmin. "A View on the Syntactical Relations." *American Journal of Linguistics* 3, no. 2 (2014): 41-45.

[21] AL-Khawaldeh, Fatima T. "Answer Extraction for Why Arabic Questions Answering Systems: EWAQ." *World of Computer Science & Information Technology Journal* 5, no. 5 (2015).

[22] Allam, Ali Mohamed Nabil, and Mohamed Hassan Haggag. "The question answering systems: A survey." International Journal of Research and Reviews in Information Sciences (IJRRIS) 2, no. 3 (2012): 211-220.

[23] Almarwani, Nada, and Mona Diab. "Arabic Textual Entailment with Word Embeddings." In Proceedings of the Third Arabic Natural Language Processing Workshop, pp. 185-190. 2017.

[24] Aronoff, Mark, and Kirsten Fudeman. What is morphology?. Vol. 8. John Wiley & Sons, 2011.

[25] Aunimo, Lili. "Methods for Answer Extraction in Textual Question Answering." (2007).

[26] Bakari, Wided, Omar Trigui, and Mahmoud Neji. "Logic-based approach for improving Arabic question answering." In Computational Intelligence and Computing Research (ICCIC), 2014 IEEE International Conference on, pp. 1-6. IEEE, 2014.

[27] BEKHTI, SMAN, and M. A. R. Y. A. M. AL-HARBI. "AQuASys: A Question-Answering System For Arabic." In WSEAS International Conference. Proceedings. Recent Advances in Computer Engineering Series, no. 12. WSEAS, 2013

[28] Benajiba, Yassine, Paolo Rosso, and AbdelouahidLyhyaoui. "Implementation of the ArabiQA Question Answering System's components." In Proc. Workshop on Arabic Natural Language Processing, 2nd Information Communication Technologies Int. Symposium, ICTIS-2007, Fez, Morroco, April, pp. 3-5. 2007.

[29] Benajiba, Yassine, and Paolo Rosso. "Arabic question answering." Diploma of advanced studies. Technical University of Valencia, Spain (2007).

[30] Bhaskar, Pinaki, Somnath Banerjee, Partha Pakray, Samadrita Banerjee, Sivaji Bandyopadhyay, and Alexander Gelbukh. "A hybrid question answering system for Multiple Choice Question (MCQ)." In *CEUR-WS*. 2013.

[31] Black, William, SabriElkateb, Horacio Rodriguez, Musa Alkhalifa, PiekVossen, Adam Pease, and Christiane Fellbaum. "Introducing the Arabic wordnet project." In Proceedings of the Third International WordNet Conference, pp. 295-300. 2006.

[32] Brini, Wissal, MariemEllouze, Omar Trigui, Slim Mesfar, H. L. Belguith, and Paolo Rosso. "Factoid and definitional Arabic question answering system." Post-Proc. NOOJ-2009, Tozeur, Tunisia, June (2009): 8-10.

[33] Buckwalter, Tim. Buckwalter Arabic Morphological Analyzer Version 2.0. Linguistic Data Consortium, University of Pennsylvania, 2002. LDC Cat alog No.: LDC2004L02. ISBN 1-58563-324-0, 2004.

[34] Castillo, Julio Javier. "A WordNet-based semantic approach to textual entailment and cross-lingual textual entailment." *International Journal of Machine Learning and Cybernetics* 2, no. 3 (2011): 177-189.

[35] Castillo, Julio, and Paula Estrella. "SAGAN: an approach to semantic textual similarity based on textual entailment." In Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation, pp. 667-672. Association for Computational Linguistics, 2012.

[36] Cavalli-Sforza, Violetta, Hind Saddiki, Karim Bouzoubaa, LahsenAbouenour, Mohamed Maamouri, and Emily Goshey. "Bootstrapping a WordNet for an Arabic dialect from other WordNets and dictionary resources." In 2013 ACS International Conference on Computer Systems and Applications (AICCSA), pp. 1-8. IEEE, 2013.

[37] Changqing, Yao, Jiangli Liu, Yongping Du. “Textual Entailment Recognition Based on Effective Text Features” Journal of Convergence Information Technology vol7,issue13.37, July 2012.

[38] Chomsky, Noam. "Syntactic structures. The Hague: Mouton.. 1965. Aspects of the theory of syntax." Cambridge, Mass.: MIT Press.(1981) Lectures on Government and Binding, Dordrecht: Foris.(1982) Some Concepts and Consequences of the Theory of Government and Binding. LI Monographs 6 (1957): 1-52

[39] Cristianini, Nello, and John Shawe-Taylor. "An introduction to support vector machines." (2000).

[40] Dagan, Ido, and Oren Glickman. "Probabilistic textual entailment: Generic applied modeling of language variability." (2004).

[41] Dagan, Ido, Oren Glickman, and Bernardo Magnini. "The PASCAL recognising textual entailment challenge." In *Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising tectual entailment*, pp. 177-190. Springer, Berlin, Heidelberg, 2006.

[42] Darwish, Kareem. "Named entity recognition using cross-lingual resources: Arabic as an example." In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, pp. 1558-1567. 2013.

[43] Darwish, Kareem, and Ossama Emam. "Retrieving Arabic Printed Document: a Survey." ICUDL, Bibliotheca Alexandrina, Alexandria, Egypt.2006.

[44] Darwish, Kareem, and Walid Magdy. Arabic information retrieval. Now Publishers, 2014.

[45] El-Defrawy, Mahmoud, Yasser El-Sonbaty, and Nahla Belal. "Enhancing root extractors using light stemmers." In Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation: Posters, pp. 157-166. 2015.

[46] Ezzeldin, Ahmed, and Mohamed Shaheen. "A survey of Arabic question answering: Challenges, tasks, approaches, tools, and future trends." In Proceedings of The 13th International Arab Conference on Information Technology (ACIT 2012), pp. 1-8. 2012.

[47] Ezzeldin, Ahmed Magdy, Mohamed Hamed Kholief, and Yasser El-Sonbaty. "ALQASIM: Arabic language question answer selection in machines." In International Conference of the Cross-Language Evaluation Forum for European Languages, pp. 100-103. Springer, Berlin, Heidelberg, 2013.

[48] Fareed, Noha S., Hamdy M. Mousa, and Ashraf B. Elsisi. "Enhanced semantic Arabic Question Answering system based on Khoja stemmer and AWN." In Computer Engineering Conference (ICENCO), 2013 9th International, pp. 85-91. IEEE, 2013.

[49] Farghaly, Ali, and Khaled Shaalan. "Arabic natural language processing: Challenges and solutions." ACM Transactions on Asian Language Information Processing (TALIP) 8, no. 4 (2009): 14.

[50] Fazza, A., James, D., Zuhair, A., & Keeley, A. (2012). Arabic Word Semantic Similarity. Proceedings of World Academy of Science, Engineering and Technology. No. 70. World Academy of Science, Engineering and Technology.

[51] Ferrández, Óscar, Rafael Muñoz, and Manuel Palomar. "TE4AV: Textual entailment for answer validation." In Natural Language Processing and Knowledge Engineering, 2008. NLP-KE'08. International Conference on, pp. 1-8. IEEE, 2008.

[52] Gaona, Miguel Angel Ríos, Alexander Gelbukh, and Sivaji Bandyopadhyay. "Recognizing textual entailment using a machine learning approach." In Mexican International Conference on Artificial Intelligence, pp. 177-185. Springer, Berlin, Heidelberg, 2010.

[53] George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, Katherine J. Miller, "Introduction to WordNet: An On-line Lexical Database", International Journal of Lexicography, Vol. 3, No. 4, pp. 235-244, 1990.

[54] Gonçalves, Patricia Nunes, and AntónioHortaBranco. "A Comparative Evaluation of QA Systems over List Questions." In International Conference on Computational Processing of the Portuguese Language, pp. 115-121. Springer International Publishing, 2016.

[55] Goweder, ABDUELBASET M., IBRAHIM A. Almerhag, and ANES A. Enakoa. "Arabic Broken Plural Recognition Using a Machine Translation Technique." (2008).

[56] Gross, Annegret M. "Information Retrieval in Arabic: An evaluation of three multilingual search engines on their capabilities in dealing with Arabic search queries." (2012).

[57] Gupta, Poonam, and Vishal Gupta. "A survey of text question answering techniques." International Journal of Computer Applications 53, no. 4 (2012): 1-8.

[58] Habash, Nizar. "Arabic morphological representations for machine translation." In *Arabic computational morphology*, pp. 263-285. Springer, Dordrecht, 2007.

[59] Habash, Nizar, Owen Rambow, and Ryan Roth. "MADA+ TOKAN: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization." In Proceedings of the 2nd international conference on Arabic language resources and tools (MEDAR), Cairo, Egypt, vol. 41, p. 62. 2009.

[60] Haggag, Mohamed H., Marwa MA ELFattah, and Ahmed Mohammed Ahmed. "Different Models and Approaches of Textual Entailment Recognition." International Journal of Computer Applications 142, no. 1 (2016).

[61] Hammo, Bassam, Hani Abu-Salem, and Steven Lytinen. "QARAB: A question answering system to support the Arabic language." In Proceedings of the ACL-02 workshop on Computational approaches to semitic languages, pp. 1-11. Association for Computational Linguistics, 2002.

[62] Handayani, AnikNur. "Collaborative e-learning system utilizing Question Answering system with domain knowledge and answer quality predictor." PhD diss., 2014.

[63] Herrera, Jesús, Anselmo Penas, and Felisa Verdejo. "Textual entailment recognition based on dependency analysis and wordnet." In *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, pp. 231-239. Springer, Berlin, Heidelberg, 2006.

[64] H. G_omez-Adorno, D. Pinto, and V. A. Darnes, "Question answering system for reading comprehension tests," Pattern Recognit., vol. 7914, pp. 354–363, 2013.

[65] Hijjawi, Mohammad and Elsheikh Yousef. Arabic Language Challenges in Text Based Conversational Agents ComparedToTheEnglishLanguage". International Journal of Computer Science & Information Technology (IJCSIT) Vol 7, No 3, June 2015.

[66] Jurafsky, Dan, and James H. Martin. Speech and language processing. Pearson, 2014.

[67] Kadri, Youssef, and Jian-Yun Nie. "Effective stemming for Arabic information retrieval." In The Challenge of Arabic for NLP/MT, Intl Conf. at the BCS, pp. 68-74. 2006

[68] Kanaan, Ghassan, AwniHammouri, Riyad Al-Shalabi, and MajdiSwalha. "A new question answering system for the Arabic language." American Journal of Applied Sciences 6, no. 4 (2009): 797.

[69] Kanaan, Raed, and GhassanKanaan. "AN IMPROVED ALGORITHM FOR THE EXTRACTION OF TRILITERAL ARABIC ROOTS." European Scientific Journal 10, no. 3 (2014).

[70] Khader, Mariam, Arafat Awajan, and Akram Alkouz. "Textual Entailment for Arabic Language based on Lexical and Semantic Matching." International Journal of Computing & Information Sciences 12, no. 1 (2016): 67.

[71] Khalil, Hussein, and Taha Osman. "Challenges in Information Retrieval from Unstructured Arabic Data." In Computer Modelling and Simulation (UKSim), 2014 UKSim-AMSS 16th International Conference on, pp. 456-461. IEEE, 2014.

[72] Khoja, Shereen, and Roger Garside. "Stemming arabic text." Lancaster, UK, Computing Department, Lancaster University (1999).

[73] Ko, Jeongwoo, Luo Si, and Eric Nyberg. "Combining evidence with a probabilistic framework for answer ranking and answer merging in question answering." Information processing & management 46, no. 5 (2010): 541-554.

[74] Kozareva, Zornitsa, and Andrés Montoyo. "MLENT: The machine learning entailment system of the University of Alicante." In Proc. of 2nd PASCAL Challenges Workshop on Recognising Textual Entailment, Venice, Italy. 2006.

[75] Kozareva, Zornitsa, and Andrés Montoyo. "The role and resolution of textual entailment in natural language processing applications." In International Conference on Application of Natural Language to Information Systems, pp. 186-196. Springer, Berlin, Heidelberg, 2006.

[76] Kurdi, Heba, Sara Alkhaider, and Nada Alfaifi. "Development and evaluation of a web based question answering system for Arabic language." *Computer Science & Information Technology (CS & IT)* 4, no. 02 (2014): 187-202.

[77] LahsenAbouenour. "Three-levels Approach for Arabic Question Answering Systems." PhD diss., EcoleMohammadiad'Ingénieurs, 2014.

[78] Lampert, Andrew. "A quick introduction to question answering." Dated December (2004).

[79] Lancioni, Giuliano, Ivana Pepe, Alessandra Silighini, Valeria Pettinari, IlariaCicola, Leila Benassi, and Marta Campanelli. "Arabic Meaning Extraction through Lexical Resources: A General-Purpose Data Mining Model for Arabic Texts."The Third International Conference on Advances in Information Mining and Management .IMMM. 2013.

[80] Leacock, Claudia, and Martin Chodorow. "Combining local context and WordNet similarity for word sense identification." *WordNet: An electronic lexical database* 49, no. 2 (1998): 265-283.

[81] Lebedeva, Olga, and Zaitseva, Larissa, “Question Answering Systems in Education and their Classifications”, In Joint International Conference on Engineering Education & International Conference on Information Technology, 359-366(2014).

[82] Lieber, Rochelle. Introducing morphology. Cambridge University Press, 2009.

[83] Lin, Chin-Yew. "Rouge: A package for automatic evaluation of summaries." Text Summarization Branches Out (2004).

[84] Li, Xin, and Dan Roth. "Learning question classifiers." In Proceedings of the 19th international conference on Computational linguistics-Volume 1, pp. 1-7. Association for Computational Linguistics, 2002.

[85] Maytham Alabbas. Textual Entailment for Modern Standard Arabic. PhD thesis, University of MANCHESTER, UK, 2013.

[86] McDonald, Ryan, and Fernando Pereira. "Online learning of approximate dependency parsing algorithms." In *11th Conference of the European Chapter of the Association for Computational Linguistics*. 2006.

[87] Mehdad, Yashar, Matteo Negri, and Marcello Federico. "Towards cross-lingual textual entailment." In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp. 321-324. Association for Computational Linguistics, 2010.

[88] Mesfar, Slim. "Named entity recognition for arabic using syntactic grammars." In Natural Language Processing and Information Systems, pp. 305-316. Springer Berlin Heidelberg, 2007.

[89] Mishra, Amit, and Sanjay Kumar Jain. "A survey on question answering systems with classification." Journal of King Saud University-Computer and Information Sciences (2015).

[90] Mohammed, F. A., Khaled Nasser, and H. M. Harb. "A knowledge based Arabic question answering system (AQAS)." ACM SIGART Bulletin 4, no. 4 (1993): 21-30.

[91] Mohammed, Nababteh, and Deri Mohammed. "Experimental Study of Semantic Similarity Measures on Arabic WordNet." International Journal of Computer Science and Network Security (IJCSNS) 17, no. 2 (2017): 131.

[92] Moldovan, Dan, SandaHarabagiu, Marius Pasca, Rada Mihalcea, Roxana Girju, Richard Goodrum, and Vasile Rus. "The structure and performance of an open-domain question answering system." In Proceedings of the 38th Annual Meeting on Association for Computational Linguistics, pp. 563-570. Association for Computational Linguistics, 2000.

[93] Monz, Christof. From document retrieval to question answering. Institute for Logic, Language and Computation, 2003.

[94] Moruz, M. A. "Predication Driven Textual Entailment." PhD diss., Ph. D. thesis,“Alexandru Ioan Cuza” University, Faculty of Computer Science, Iasi, 2011.

[95] Nabil, Mohamed, Ahmed Abdelmegied, Yasmin Ayman, Ahmed Fathy, Ghada Khairy, Mohammed Yousri, Nagwa M. El-Makky, and Khaled Nagi. "AlQuAnS-An Arabic Language Question Answering System." In *KDIR*, pp. 144-154. 2017.

[96] Orasan, Constantin, Dan Cristea, Ruslan Mitkov, and António Horta Branco. "Anaphora Resolution Exercise: an Overview." In *LREC*. 2008.

[97] Pakray, Partha, Alexander Gelbukh, and Sivaji Bandyopadhyay. "A syntactic textual entailment system based on dependency parser." In International Conference on Intelligent Text Processing and Computational Linguistics, pp. 269-278. Springer, Berlin, Heidelberg, 2010.

[98] Pakray, Partha, Pinaki Bhaskar, Somnath Banerjee, Bidhan Chandra Pal, Sivaji Bandyopadhyay, and Alexander F. Gelbukh. "A Hybrid Question Answering System based on Information Retrieval and Answer Validation." In CLEF (Notebook Papers/Labs/Workshop). 2011.

[99] Peñas, Anselmo, Eduard H. Hovy, Pamela Forner, Álvaro Rodrigo, Richard FE Sutcliffe, Corina Forascu, and Caroline Sporleder. "Overview of QA4MRE at CLEF 2011: Question Answering for Machine Reading Evaluation." In *CLEF (Notebook Papers/Labs/Workshop)*, pp. 1-20. 2011.

[100] Peñas, Anselmo, Eduard Hovy, Pamela Forner, Álvaro Rodrigo, Richard Sutcliffe, and RoserMorante. "QA4MRE 2011-2013: Overview of question answering for machine reading evaluation." In International Conference of the Cross-Language Evaluation Forum for European Languages, pp. 303-320. Springer Berlin Heidelberg, 2013.

[101] Pedersen, Ted, Siddharth Patwardhan, and Jason Michelizzi. "WordNet:: Similarity: measuring the relatedness of concepts." In *Demonstration papers at HLT-NAACL 2004*, pp. 38-41. Association for Computational Linguistics, 2004.

[102] Raj, P. C. "Architecture of an Ontology-Based Domain-Specific Natural Language Question Answering System." arXiv preprint arXiv:1311.3175 (2013).

[103] Ray, Santosh K., and Khaled Shaalan. "A review and future perspectives of arabic question answering systems." IEEE Transactions on Knowledge and Data Engineering 28, no. 12 (2016): 3169-3190.

[104] Ren, Han, Donghong Ji, and Jing Wan. "WHU at TAC 2009: A Tri-categorization Approach to Textual Entailment Recognition." In TAC. 2009.

[105] Romeo, Salvatore, Giovanni Da San Martino, Yonatan Belinkov, Alberto Barrón-Cedeño, Mohamed Eldesouki, Kareem Darwish, Hamdy Mubarak, James Glass, and Alessandro Moschitti. "Language processing and learning models for community question answering in Arabic." *Information Processing & Management* (2017).

[106] Rosso, Paolo, YassineBenajiba, and AbdelouahidLyhyaoui. "Towards an Arabic question answering system." In Proc. 4th Conf. on Scientific Research Outlook & Technology Development in the Arab world, SROIV, Damascus, Syria, pp. 11-14. 2006.

[107] Ryding, Karin C. "Modern Standard Arabic." Cambridge University Pres, UK (2005).

[108] Shaalan, Khaled. "Rule-based approach in Arabic natural language processing." The International Journal on Information and Communication Technologies (IJICT) 3, no. 3 (2010): 11-19.

[109] Shaheen, Mohamed, and Ahmed MagdyEzzeldin. "Arabic question answering: Systems, resources, tools, and future trends." Arabian Journal for Science and Engineering 39, no. 6 (2014): 4541-4564.

[110] Sherkat, Ehsan, and Mojgan Farhoodi. "A hybrid approach for question classification in Persian automatic question answering systems." In *Computer and Knowledge Engineering (ICCKE), 2014 4th International eConference on*, pp. 279-284. IEEE, 2014.

[111] Shivhare, Himanshu, Parul Nath, and Anusha Jain. "Semi Cognitive approach to RTE 6-Using FrameNet for Semantic Clustering." In TAC. 2010.

[112] Slimani, Thabet. "Description and evaluation of semantic similarity measures approaches." arXiv preprint arXiv:1310.8059 (2013).

[113] Soubbotin, Martin M., and Sergei M. Soubbotin. "Patterns of potential answer expressions as clues to the right answers." In TREC. 2001.

[114] Taghva, Kazem, Rania Elkhoury, and Jeffrey Coombs. "Arabic stemming without a root dictionary." In *Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on*, vol. 1, pp. 152-157. IEEE, 2005

[115] Téllez-Valero, Alberto, Manuel Montes-y-Gómez, Luis Villasenor-Pineda, and AnselmoPenas. "Improving question answering by combining multiple systems via answer validation." In Computational Linguistics and Intelligent Text Processing, pp. 544-554. Springer Berlin Heidelberg, 2008.

[116] Trigui, Omar, Lamia Hadrich Belguith, Paolo Rosso, Hichem Ben Amor, and Bilel Gafsaoui. "Arabic QA4MRE at CLEF 2012: Arabic Question Answering for Machine Reading Evaluation." In CLEF (Online Working Notes/Labs/Workshop). 2012.

[117] van Zaanen, M. (2002). *Bootstrapping Structure into Language: Alignment-Based Learning*. PhD thesis, University of Leeds, Leeds, UK, January 2002.

[118] Vapnik, Vladimir. *The nature of statistical learning theory*. Springer-Verlag Berlin, Heidelberg, 1995.

[119] Voorhees, Ellen M., and Donna Harman. "The text REtrieval conference (TREC): History and plans for TREC-9." In ACM SIGIR Forum, vol. 33, no. 2, pp. 12-15. ACM, 1999.

[120] Wang, Rui, and Günter Neumann. "Recognizing textual entailment using a subsequence kernel method." In AAAI, vol. 7, pp. 937-945. 2007.

[121] Wu, Zhibiao, and Martha Palmer. "Verbs semantics and lexical selection." In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pp. 133-138. Association for Computational Linguistics, 1994.

[122] Zhang, Kaizhong, and Dennis Shasha. "Simple fast algorithms for the editing distance between trees and related problems." SIAM journal on computing 18, no. 6 (1989): 1245-1262.

[123] Zhang, Yuan, Chengtao Li, Regina Barzilay, and Kareem Darwish. "Randomized greedy inference for joint segmentation, POS tagging and dependency parsing." In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 42-52. 2015.

[124] Zheng, Zhiping. "AnswerBus question answering system." In Proceedings of the second international conference on Human Language Technology Research, pp. 399-404. Morgan Kaufmann Publishers Inc., 2002.

[125] Zitouni, Imed, ed. Natural language processing of semitic languages. Springer, 2014.

[126] Zmai, Aqil M., and Nouf A. Alshenaifi. "Answering arabic why-questions: Baseline vs. rst-based approach." *ACM Transactions on Information Systems (TOIS)* 35, no. 1 (2016): 6.