A Text Mining Approach for Arabic Question Answering Systems

Jawad Sadek

Ph.D. Thesis **2014**

A Text Mining Approach for Arabic Question Answering Systems

Jawad Sadek

School of Computing, Science and Engineering
College of Science and Technology
University of Salford, Salford, UK

Submitted in Partial Fulfilment of the Requirements of the Degree of Doctor of Philosophy, November 2014

Declaration

I hereby confirm that this thesis represents my own work. No portion of the work referred to in the thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Table of Contents

List of Tables

List of Figures

Acknowledgements

I would like to thank my parents for their support, caring and believing in me.

I would like to show gratitude and deep respect to my supervisor Prof. Farid Meziane for his continuous support and encouragement, thoughtful feedback and excellent guidance.

I would like to thank Dr. Fairouz Chakkour for her guidance in the beginning of the research. I also would like to thank Dr. Nadia Haskkour for helping out and giving advice in some issues related to this research.

Many thanks go to my friends and colleagues for their love and support.

Special thanks to the University of Salford for its support to conduct this research.

Abstract

As most of the electronic information available nowadays on the web is stored as text, developing Question Answering systems (QAS) has been the focus of many individual researchers and organizations. Relatively, few studies have been produced for extracting answers to "*why*" and "*how to*" questions. One reason for this negligence is that when going beyond sentence boundaries, deriving text structure is a very time-consuming and complex process. This thesis explores a new strategy for dealing with the exponentially large space issue associated with the text derivation task. To our knowledge, to date there are no systems that have attempted to addressing such type of questions for the Arabic language.

We have proposed two analytical models; the first one is the *Pattern Recognizer* which employs a set of approximately 900 linguistic patterns targeting relationships that hold within sentences. This model is enhanced with three independent algorithms to discover the causal/explanatory role indicated by the justification particles. The second model is the *Text Parser* which is approaching text from a discourse perspective in the framework of Rhetorical Structure Theory (RST). This model is meant to break away from the sentence limit. The *Text Parser* model is built on top of the output produced by the *Pattern Recognizer* and incorporates a set of heuristics scores to produce the most suitable structure representing the whole text.

The two models are combined together in a way to allow for the development of an Arabic QAS to deal with *"why"* and *"how to"* questions. The *Pattern Recognizer* model achieved an overall *recall* of 81% and a *precision* of 78%. On the other hand, our question answering system was able to find the correct answer for 68% of the test questions. Our results reveal that the justification particles play a key role in indicating intrasentential relations.

Chapter 1

Introduction

1.1 Arabic NLP

Arabic is the sixth most widely spoken language in the world and is ranked fifth among the most influential languages in the world according to research performed by George Weber (1997). He stated that *"Arabic is the only language apart from English and French that is used in an international field"*. This is mainly attributed to its political and economic significance in addition to being the language of worship for over 1.5 billion Muslims. Moreover, a recent report published by the United Nations revealed that the rapid rising of the Internet use in the Middle East has resulted in Arabic becoming the fastest-growing language on the Internet in the past decade (The Arab Knowledge Report, 2011).

Arabic content on the Web has seen a phenomenal growth in the past few years, and it has become very difficult to manually extract information from these resources, particularly from unstructured texts. Consequently, all tasks of Natural Language Processing (NLP) will become increasingly essential to make Information Retrieval (IR), Text Mining (TM), text categorization, automatic summarization, machine translation and question answering systems available to the Arab user.

Compared to the other languages, there are relatively few studies developed to manipulate knowledge encoded in the Arabic language. This is mainly due to the challenges and complexities present in Semitic languages like Arabic which are known to be highly derivational and inflectional (Kadri and Benyamina, 1992).

The Arabic morphology is very rich. Conjunctions, definite articles, particles and other prefixes can be attached to the beginning of a word, and a large number of suffixes can be attached to the end. Both prefixes and suffixes are allowed to be combined and present at the same time. This generates a huge number of different forms for a given root.

Diacritics also contribute to the variability of Arabic words adding confusion to NLP applications. Indeed, the same word with different diacritics can express different meaning, for example, مالْ"money" and مالَ"incline". However, diacritics are only found in specialized contexts such as dictionaries, children's books, and the Quran.

The irregular syntactic form of Arabic sentences is an additional problem which results in great flexibility in changing the subject and verb positions. Consider for example the mutual swapping of the words "ضرب" and "الرجل" in sentences (1) and (2) and yet they have the same meaning.

(1) ضرب الرجل الولد

(2) الرجل ضرب الولد

"The man hit the boy"

Another reason why Arabic NLP lags behind is the lack of mature tools and knowledge bases resources available for Arabic unlike the other languages which benefited from the existence of huge corpora and annotated Treebanks for training.

1.2 Answer Extraction from Textual Resources

There is a high demand for systems that could return a precise answer to a user's query and avoid the thousands of links returned by traditional search engines. In the NLP field, these systems are referred to as QA systems and these could be developed for open or specific domains. However, current QA systems involve intensive computing and often fail to match the speed of current search engines.

QA systems are known to be of great importance in many real life application areas. For example, in the field of medicine, physicians are unable to respond to all patient queries within the required time, leaving most of the questions unanswered. Hence, a QA clinical

system would be capable of returning answers based on existing medical research reports (Niu and Hirst, 2009). QA systems have also been explored for educational packages by replying to quick questions posed by users who simply need a fast reference such as the publication date of a certain book or the population of a city (Aria and Handayani, 2012). Furthermore, QA systems were incorporated into decision support systems (Yang et al., 2014), business intelligence (Choi et al., 2011) and interactive QA systems where a Chabot-based interface enables conducting a conversation that attempts to emulate human dialogue (Wang and Petrina, 2013).

A variety of approaches to QA have been investigated in TREC-QA evaluation campaigns. Answer classes targeted by most QA systems were of the factoid type generally seeking short fact based answers (e.g. names, dates, and places). In QA systems involving factoid questions, Named Entity recognition can make a substantial contribution to identifying potential answers in a source document where the answer units are no more than few words expressed in the form of a noun phrase as shown in Table 1-1.

Question	**Named Entity**
Who/whose	Person
When	Time, Date
Where	Location
How much	Quantity
How many	Number
How Long	Duration

Table 1-1: Question types and their corresponding Named Entity reference.

Recently, a number of systems were implemented where the focus has shifted away from fact-based questions to handling questions requiring non-factoid and more complex answers such as causation, manner or reason questions. Unlike factoid QA, these systems are expected to return answers in the form of a meaningful discourse segment (i.e. sentence, multiple sentences and paragraph).

1.3 Motivation

There are very few QA systems specifically developed for the Arabic language and those developed focused on factoid questions that can be answered with relatively little linguistic knowledge (Mohammed et al., 1993; Hammou et al., 2002; Kanaan et al., 2009). Like the case of TREC participants, question types that require long and procedural answers such as "*why*" and "*how to*" was beyond the scope of those systems.

However, most questions that people want answers for are not factoid questions. Statistics showed that questions starting with "*why*" and "*how*" are quite frequently issued by users on social media such as *Yahoo! answers*[1]. Verberne (2010) reported that Microsoft's Web Search Click Data, a collection of queries from US users entered into Microsoft Live search engine in the summer 2006, contained 86,391 queries starting with *wh-question (who, what, which, where, when, how and why)*. Of these, queries starting with *"how"* and *"why"* were by far the most frequent (61%). Yet out of the *"how"* questions approximately 76% were of the type *"how to"* while the rest were subtypes that referred to quantity questions (*how much, how many, how long,* etc.).

To the best of our knowledge no previous Arabic QA system was developed to specifically answer "*why*" and "*how to*" questions in spite of their frequency and significance in a wide range of disciplines (clinical, education, social communities etc). It is also the case that the task of automatic extraction of *Causal* relations is still absent in the Arabic research area. Thus, novel approaches need to be devised to meet this shortcoming in the Arabic NLP field and this was our main motivation to develop the work presented in this thesis.

1.4 Contribution

As pointed out in the previous sections, different techniques are needed to handle non-factoid questions whose corresponding answers often span multiple sentences that comprise discourse relations such as cause, motivation, purpose and explanation. One issue here is that these relations are often expressed implicitly using verbal or non-verbal cue words. What makes this research more challenging is that recognizing the answer boundaries involves conducting

[1] http://answers.yahoo.com

advanced analysis (e.g. syntactic and semantic). All these issues make the task of finding the exact answers to "*why*" and "*how to*" questions a very challenging problem.

There are some studies (Breck et al., 2000; Bernardi et al., 2003) that investigated the task of locating exact answers to non-factoid questions; they reported that such type of questions require fine-grained text analysis and reasoning capabilities. Moreover, they suggested that the wise exploitation of linguistic knowledge (i.e. the knowledge about discourse structure) would allow QA systems to answer "*why"* questions.

In this research, "*why"* and "*how to*" questions are defined as an interrogative sentence in which the interrogative nouns لماذا "*why"* - كيف "*how to"* (or a synonymous word or phrase) occurs in the initial position. In this context, "*Why*" questions enquire about events or facts that explains why something occurred rather than something else whereas "*how to*" questions enquire about the manner in which something is done.

The main contribution of this study is to carry out an extensive Arabic text analysis in order to devise a set of linguistic patterns which are able to indicate the presence of causation/explanation information in sentences from open domain texts. The constructed patterns will be developed predominantly to locate relations within sentences (intrasentential relations) and this will be combined with a linguistically aware model that discovers relations among sentences (intersentential relations).

For the purpose of finding causation and explanation across sentences, we will employ the Rhetorical Structure Theory (RST) that many studies have shown to be a very effective discourse analysis approach for many computational linguistics applications such as (text generation, text summarization and machine translation). In his work on rhetorical parsing of unrestricted English texts, Marcu (2000b) examined a great number of connectives such as therefore, although, in contrast etc; he stated *"it is likely that connectives can be used in order to determine rhetorical relations that hold between elementary units"*. In this study we exploit the knowledge of the connectives and cohesion in the Arabic text to posit suitable rhetorical relations.

1.5 Research Questions

Our research objectives focus on answering the following two questions:

- ***Is it possible for hand-crafted patterns to convey information from open domain text using a subset of NLP techniques?***

  Given that linguistic knowledge is expensive, we identify a set of linguistic patterns based on syntactic and linguistic features which comprises a combination of cue words and Part Of Speech labels (POS) that tend to appear in causal and explanatory sentences.

  To fulfil this aim, we first investigate existing literature on the subject to explore the linguistic devices identified by researchers whose function is primarily to indicate causation and explanation. Arabic text analysis will then be carried out to establish which syntactic features truly appear to be relevant for detecting causation and explanation at the sentence level.

- ***To what extent can discourse analysis help in selecting answers to "why" and "how to" questions for the Arabic language?***

  To be able to extract meaningful answers to non-factoid questions from a text, it is crucial to have knowledge about its structure. The structure of text can be visualized by annotating the text with intrasentential/intersentential relations. This annotated text can then be queried for questions correlate with specific type of relations.

  Apparently, the task of the automatic derivation of discourse structure at all text levels requires huge computing power. Therefore, a more practical approach is required to tackle this problem.

  Obviously, considering relations spanning over only individual sentences one at a time is more computationally efficient than considering the whole text. Furthermore, Arabic writers prefer the use of grouped and large grammatical chunks and it is rare that an Elementary Discourse Unit (EDU) from a sentence has a relation with a part outside the sentence.

  The approach we adopt in this study splits the process of text analysis into two different models. First, we create the *Pattern Recognizer* model for causal and explanatory knowledge acquisition within sentences based on a set of linguistic

patterns. Second, we build the *Text Parser* model that would hypothesize a list of rhetorical relations which hold among sentences. This model incorporates the intrasentential information provided by the *Pattern Recognizer* and produces the most suitable structure representing the whole text.

1.6 Thesis Structure

The current thesis is structured into chapters that describe various aspects of this research. In the following subsection, we summarize each of these chapters.

Chapter 1 introduces the issue of answer extraction and QA systems including both factoid and non-factoid questions. It also introduces motivations and significance of the study. After that, research questions are presented followed by an overview of the thesis.

Chapter 2 consists of two main sections. The first part presents a brief history of computer based question answering systems and the role played by the Text Retrieval Conference (TREC) in the development of these systems. It also provides an overview of the approaches adopted for processing the three parts of QA systems namely question processing, passage retrieval and answer processing. The second part of this chapter is devoted to describing the relevant work of other researchers in the field of answering non-factoid questions and in the area of extracting semantic relations from text.

Chapter 3 investigates the first research question by describing the procedures adopted for extracting potentially syntactic features and relevant coherence markers that would lead to constructing a set of linguistic patterns.

Chapter 4 contains a novel contribution to the field of Arabic text structure derivation. This chapter answers the second research question. The chapter starts with a brief explanation of the framework used in this study, RST, along with a general review of the automatic text derivation systems. Next, it describes the proposed methodology that attempts to deal with the problem of computational complexity associated with the text derivation process.

Chapter 5 illustrates the infrastructure of the question answering system developed in this research and how we apply the two models proposed throughout the previous chapters. The chapter also studies several techniques introduced by researchers in the field of Arabic

Information Retrieval. These techniques - *Normalization*, *Stemming*, *Stop-words removal* - aim to handle challenges raised when processing the Arabic language and are essentials tools for the implementation of different components in our system.

Chapter 6 outlines the experiments conducted with the participation of human judges to observe the effectiveness of the individual and overall performance of the system. It analyzes the performance of the *Pattern Recognizer* model under different conditions using the *recall*, *precision* and F score measures. Moreover, the chapter shows the experiment performed to evaluate the system efficiency in finding answers to "*why*" and "*how to*" questions. Finally, the chapter concludes this thesis by stating the main results obtained in this research followed by recommendations for future work.

Chapter 2

Literature Review and Related Work

2.1 Overview of QA Systems

2.1.1 A Brief History

The first QA systems emerged in the early 1960s and 1970s as natural language interfaces for databases containing specific information about a topic, such as the BASEBALL (Green et.al, 1961) and LUNAR (Woods et.al, 1972) systems that operated on very restricted domains. The former answered questions about the United States baseball league during a single season and the latter replied to questions on the rocks returned from the moon by the Apollo moon missions. The questions presented to these systems were usually analyzed using linguistic knowledge to produce a canonical form, which was then used to construct a standard database query.

Computer systems capable of holding a meaningful conversation are usually referred to as dialogue systems and have emerged by the end of the 1960s. One of the earliest and best known of these Artificial Intelligence dialogue systems is the ELIZA system (Weizenbaum, 1966) that provided a psychological conversation in which patients were able to converse with ELIZA as in an initial psychiatric interview. Two other dialogue systems were developed later; SHRDLU system (Winogra, 1972) that answered questions about different states in a Toy World, and GUS system (Bobrow et al., 1977) which was designed to simulate a travel advisor and had access to a database containing limited information about airline flight times.

QA systems took a further step with the development of the computational linguistics domain, which aimed to develop automated software capable of understanding the meaning of texts.

QUALM system (Lehnert, 1977) used knowledge bases and rule-based reasoning (Schank and Abelson, 1977) to build a system able to answer comprehension tests.

Another example of such systems was Unix Consultant (Wilensky, 1982) that was designed to answer technical questions about the UNIX operating system. SCISOR (Jacobs and Rau, 1990) focused on the question answering task more than information retrieval; it combined NLP, knowledge representation, and information retrieval techniques with lexical analysis and word-based text searches.

The MURAX system (Kupiec, 1993) was designed to extract answers from free texts rather than a structured database; these questions appear in the general-knowledge "Trivial Pursuit" board game. The answers were assumed to be noun phrases and thus the system provided the user with a relevant text in which noun phrases were marked.

Ask Jeeves[2] (1996) is one of the most common NLP search engines today. At its start, the *ask.com* search engine was accepting questions in a natural language and returning Web links that might contain information relevant to the answer. Ask Jeeves benefited from the use of advanced natural language processing techniques combined with data mining processing and a huge expanding knowledge base.

Another system with a different approach is the FAQFinder system (Burke et al., 1997) which attempted to analyze a user's natural language query to find a similar question that had been asked and answered previously in FAQ files.

Another important QA system was the START (SynTactic Analysis using Reversible Transformations) system (Katz, 1997) which was developed at the artificial intelligence laboratory in Massachusetts Institute of Technology (MIT). The START system analyzed English text and produces a knowledge base which incorporates, in the form of nested ternary expressions, the information found in text. A query is analyzed in the same way as assertions used to create the knowledge base.

Research into open domain QA then emerged and focused on developing question-answering system that do not rely on a knowledge base and that can extract answer from huge unstructured texts. New QA systems enhanced with NLP and IR techniques have been

[2] http://www.ask.com

developed. These were mainly motivated by the Text Retrieval Conference (TREC) for English QA systems and the Cross Language Evaluation Forum (CLEF) for multi-lingual QA system.

2.1.2 The Text REtrieval Conference

TREC is an on-going series of International conference co-sponsored by the National Institute of Standard and Technology (NIST) and the Intelligence Advanced Research Project Activity. It is focusing on a list of different IR research area called tracks.

TREC introduced the first question answering track in TREC-8 (1999). The goal of the QA track was to foster research on systems that retrieve answers rather than documents in response to a question, with particular emphasis on systems that can function in unrestricted domains (Voorhees and Tice, 2000).

In the first few editions of TREC, Mean Reciprocal Rank (MRR) was the standard TREC measure for evaluation. MRR is a score equals to the rank of the highest ranked correct answer for each question. It is calculated as follows: for each question, the reciprocal rank (RR) is 1 divided by the rank of the highest ranked correct answer or 0 if none of the responses contained a correct answer. MRR is then the average of RR over all questions.

Many QA systems from industrial and academic organizations competed against each other to answer questions that TREC provides every year. Best performing systems are then selected in each competition to present their QA approaches at the TREC conference.

A brief chronological description of the TREC is as follows:

- ***TREC-8 (1999):*** Participants received a set of short questions, and systems were asked to return a ranked list of up to five snippets that contained an answer to each question along with the Id of a document that supported the answer. Answer strings were limited to either 50 or 250 bytes in length which contained a correct answer in the context provided by the document. Human assessors read each string and made a binary decision as to whether or not the string contained an answer to the question. Twenty different participants from industrial and academic organizations received 200 questions and tested their systems on a large collection

of documents (1.5 gigabytes of text) from three different sources: TREC QA participants and NIST staff, the TREC assessors and question logs from the FAQFinder, in which each question had at least one document that explicitly answered the question (Voorhees and Tice, 1999). The best performing system for long string answers was the Textract system from Cymfony Inc (Srihari and Li, 1999) with MRR of 0.66 and for short string answers was LASSO (Moldovan et al., 1999) system from Southern Methodist University with 0.64. Most systems first attempted to classify a question according to the type of its answer as suggested by its question word. For example, a question that begins with "who" implies that a person or an organization is being sought, and a question beginning with "when" implies a time designation is needed (Voorhees and Tice, 1999). Next, the systems retrieved a small portion of the document collection. In the case of long string answers (250 bytes) standard text retrieval technology -Bag-of-words approaches- were adequate for finding answers (Allan et al., 1999; Lin and Chen, 1999; Cormack et al., 1999). But more sophisticated processing techniques such as: named entity recognition, shallow parsing and part-of-speech tagging was necessary to be employed for shorter responses (50 bytes) (Takaki, 1999; Ogden et al., 1999). This approach worked well provided the query types had enough coverage and the system could classify questions sufficiently accurately (Voorhees and Tice, 1999).

- ***TREC-9 (2000):*** In this track answers were also limited to either 50 bytes or 250 bytes and guaranteed to have an answer in the collection. However, TREC-9 used actual users' questions rather than questions constructed specifically for the track, so it was considerably harder than TREC-8 as questions tend to be more ambiguous (Voorhees and Harman, 2000). The major change between TREC-9 and TREC-8 was the creation of questions, as they were selected from query logs (Encarta and Excite log). The database was also larger consisting of 693 questions rather than 200 and a document set of all the news articles on TREC disks 1-5 (Voorhees, 2000). Five hundred questions were selected from among the candidate questions that had an answer in the document set by NIST assessors. Among twenty-eight groups, the best MRR was obtained by the FALCON system form Southern Methodist University (Harabagiu et al., 2000) for 50 bytes limit on

the length of the response with 0.58 and for 250 bytes limit with 0.76. The system was guided by three different feedback loops that tried to integrate different forms of syntactic, semantic, and pragmatic knowledge until it found an answer that provided a justification implemented as an abductive proof. As many TREC-9 systems (Ittycheriah, et al., 2000; Litkowski, 2000), it incorporated WordNet semantic net to create a large hierarchy from which it found the expected answer type and thus extracted answers after performing unifications on the semantic forms of the questions and its candidate answers.

- ***TREC 2001:*** In its third edition the QA track contained three different tasks: the list task, the context task, and the main task which was the focus of the track. The source of question set consisted of 500 questions of filtered MSNSearch logs and Ask Jeeves logs. Unlike previous years, questions were limited to no more than 50 bytes and questions were not guaranteed to have a known correct answer in the document collection allowing systems to return a response of 'NIL' to indicate their belief that no answer was present (Voorhees and Harman, 2001). Thirty-six different groups submitted to the QA track and the best performing system, TextRoller from InsightSoft-M (Soubbotin, 2001) , was able to extract a correct answer about 77% of time and an MRR of 0.68 for strict (unsupported responses counted as wrong) and MRR of 0.69 for lenient (unsupported response counted as correct) evaluation (Voorhees, 2001). Most participants used the same basic strategy; they continued to build systems that compared entities and relations between questions and candidate answers. However, many participants such as TextRoller system tend to employ a data driven approach that does not require sophisticated NLP or knowledge based analysis of question. TextRoller checked the answer candidates for predefined patterns of textual expressions to which scores were assigned beforehand. In case that no pattern was matched, the system searched the candidate answers for a lexical similarity between the question and answer snippets (Soubbotin, 2001).
- ***TREC 2002:*** Thirty-four different groups participated in this track which contained two tasks, the main task and the list task. A new document collection known as the AQUAINT Corpus of English News Text was used and comprised 1,000,000 documents and 3 gigabytes of text as the source of answers along with

500 questions drawn from MSNSearch and AskJeeves logs (mistakes fixed by NIST) (Voorhees, 2002a). As a step in improving QA, systems were required to return nothing else than one response per question or "NIL" if they believed that the collection did not contain an answer, in contrast to the previous years where systems were allowed to return text strings containing an answer. The need for this change is illustrated in the following example taken form Voorhees (2002b). The question *what river in the US is known as the Big Muddy?* yields the answer strings shown in Figure 2-1 that were judged correct. Obviously earlier responses are better than later ones.

```
the Mississippi
Known as Big Muddy, the Mississippi is the
longest
as Big Muddy , the Mississippi is the longest
messed with .  Known as Big Muddy , the Missis-
sip
Mississippi is the longest river in the US
the Mississippi is the longest river in the US,
the Mississippi is the longest
river(Mississippi)
has brought the Mississippi to its lowest
ipes.In Life on the Mississippi,Mark Twain wrote
t
Southeast;Mississippi;Mark Twain;officials began
Known; Mississippi; US,; Minnesota; Gulf Mexico
Mud Island,;Mississippi;"The;-- history,;Memphis
```

Figure 2-1: Answers strings for the question "*What river in US is known as the Big Muddy*".

Asking systems to retrieve exact answers demonstrates if they know precisely where the answer lies in such string. Another major change was the new scoring metric called confidence-weighed score. Systems were required to order their responses for the test questions from most to least confident response, so that the question for which the system felt confident was ranked first then the next most confident response and so on. The confidence-weighted score was defined in formula (2-1).

$$\frac{1}{Q} \sum_{i=1}^{Q} \frac{\text{number correct in first i ranks}}{i} \qquad (2\text{-}1)$$

The PowerAnswer system From Language Computer Corporation Moldovan et al (2002) achieved the best confidence-weighted score of 0.856 with 415 correctly answered questions. The increasing difficulty of the TREC track required systems to use more complex NLP tools. PowerAnswer tool set includes: Named Entity Recognizer, Syntactic Parser, Logic Form Transformer, Word Sense Disambiguator, and others (Moldovan, et al., 2002).

- ***TREC 2003:*** In this year, the track contained the main task and the passage task using the same source of collection answers. In the passage task, systems should return a single document not longer than 250 characters containing an answer. The main task comprised three subtasks, factoid questions, list questions, and definition question (these types will be explained in Section 2.2). The final score for a passages task was accuracy, whilst in the main task each type of question was judged and scored separately, then the final score was the weighted average of the component scores as shown in formula (2-2) (Voorhees, 2003a; Voorhees, 2003b)**.**

$$\text{FinalScore} = {}^{1}/_{2} * \text{FactoidScore} + {}^{1}/_{4} * \text{ListScore} + {}^{1}/_{4} * \text{DefScore} \quad (2\text{-}2)$$

Twenty five groups were submitted to the track, among which LLC's QA system (Harabagiu et al., 2003) from Language Computer Corporation obtained the best final score of 0.559.

- ***TREC 2004:*** Was slightly different from the previous year; the track contained one task consisting of a mix set of question types grouped into different series. Each series contained Factoid and List questions that sought information about a definition target plus one "Other" question asked for additional information about the target that was not covered by previous questions in the series (Voorhees, 2004b). Figure 2-2 shows a group of questions containing the three types of questions addressed in TREC 2004. The score for Factoid questions was the accuracy while the List and Other question were each scored using average of different computation for each type (Voorhees, 2004a). The final score was computed using formula (2-3).

$$\textbf{FinalScore} = {}^{1}/_{2} * \textbf{FactoidAccuracy} + {}^{1}/_{4} * \textbf{ListAveF} + {}^{1}/_{4} * \textbf{OtherAveF} \quad (2\text{-}3)$$

The first position was achieved by the LCC1 system with a best final score of 0.601. Generally, systems used the same techniques as were used in past years (Voorhees, 2004a).

Target ID: 65
Target string: space shuttles
65.1: LIST: What are the names of the space shuttles?
65.2: FACTOID: Which was the first flight?
65.3: FACTOID: When was the first flight?
65.4: FACTOID: When was the Challenger space shuttle disaster?
65.5: FACTOID: How many members were in the crew of the Challenger?
65.6: FACTOID: How long did the Challenger fight last before it exploded?
65.7: OTHER: Other

Figure 2-2: Example TREC 2004 question group on the topic "space shuttles".

- ***TREC 2005:*** Was held on the basis of three separate tasks: the main question answering task which was very similar to the one in the previous year except that targets could be events and nominal concepts which resulted in lower scores than last year (Voorhees, 2005). The second task was document ranking in which systems were required to return a ranked list of documents for each question; the aim was to investigate whether document retrieval techniques can help QA. The third task was relationship task to find evidence for the existence of a particular relationship within TREC-like topic statements (Voorhees and Dang, 2005). Systems were evaluated using the same methodology as in TREC 2004. The best performance again was achieved by LLC (Harabagiu et al., 2005) with a score of 0.53 employing two different systems (PowerAnswer-2) for the main task and (PALANTER) for the relationship task. They used a syntactical parser, Named Entity Recognition (NER) and a reference resolution system as tools accessible by all of the system's modules. They also took advantage of the abundance of information presented by the Internet to improve the statistical approach employed for the answer selection.

- ***TREC 2006:*** In 2006, the TREC QA had two tasks: the main task and the complex, interactive question answering task. The difference for the main task for this year was the timeframe for questions phrased in the present tense, i.e., the system was required to extract answer with the most recent information available in case more than one document in the collection was suitable, as a closer step to the real life user's requirements (Voorhees, 2006). The interactive task (ciQA) was a blend of the TREC 2005 relationship task and the TREC 2005 HARD track, the aim of the task was to incorporate a limited form of interaction with users that provided more complex information (Dang et al., 2006). The best overall score for the main task was obtained by the PowerAnswer3 system with 0.39. The improvement made from the previous year to meet the challenges of temporal constraints was the addition of the temporal resolution module. The module analyzed the target and the question together to resolve any ambiguous temporal context and used this information to create a list of reformulations of questions. At the end a voting was performed to determine which of the ambiguous target understanding reformulations had higher confidence. They also merged heuristics and machine learning algorithms for ambiguous questions where the learner's features for answer type terms included part-of-speech, lemma, head information, and named entity information (Moldovan et al., 2006).
- ***TREC 2007:*** Is the last workshop in the track series that was designed for QA systems. The track contained the same main task with a significant change in that test corpus comprised blogs documents in addition to newswire, increasing the difficulty of the task due to informal language and discourse structures nature of blogs. The scores in this task were higher after having generally declined each year since TREC 2004 (Voorhees, 2007). The Other task was complex interactive QA introduced in TREC 2006 and remained unchanged from the last year (Dang et al., 2007). PowerAnswer4 System from Lymba Corporation obtained the best overall score with 0.48. The system used a set of strategies independently or together designed to handle different types of questions. A language model was assigned for each type of questions based on features (stemmed keywords – morphological alternations for keywords and named entity tags) extracted from the questions and their answers which were judged as correct. To meet the

challenges emerged from the inclusion of blog documents (not well-formed texts - large sizes of data and organization entries), the system first performed a set of filtering steps. It parsed files to identify unique content and remove the duplicate entries, and then it used a language detection tool to remove the non-English documents, spam documents and documents containing information-deficient articles (Moldovan et al., 2007).

2.2 Classes of Question Answering

Over the years, QA systems have increased the coverage of questions they attempt to answer and become more and more complex. Hence, it is hard to classify them into well-distinguished classes. In this section we focus on the main classes of QA Systems. Generally, QA Systems can be classified into two generic categories according to the type of questions they try to answer: Fact Based Question Answering (FBQA) and non-Factoid Question Answering (NFQA).

2.2.1 Fact Based Question Answering

FBQA are closed-class types of questions seeking a single fact to be retrieved and returned to the user where systems are expected to return the exact short answer. Such types of questions can be of great importance for many applications such as in the educational domain, clinical answering systems and decision support systems. Figure 2-3 shows examples of FBQA taken from Voorhees and Harman (2000).

- How much folic acid should an expectant mother get daily?
- Who invented the paper clip?
- What university was Woodrow Wilson president of?
- Where is Rider College located?
- Name a film in which Jude Law acted.
- Where do lobsters like to live?

Figure 2-3: Examples of Factoid Question.

Another similar type is the List questions that ask for different instances of facts related to a particular kind of information and to be retrieved as a list of entities (people, places, dates, and numbers). For example, the question "*what countries are in the European Union?"* seeks for a list of country names such as "France, Germany and Italy". Voorhees (2003a) stated that "*List questions can be thought of as shorthand for asking the same factoid question multiple times; the set of answers that satisfy the factoid question is the appropriate response for the list question*".

Nearly the same approaches are used for answering both the List and Factoid questions. To guarantee that the list answers is sufficient, most TREC participants adjusted their factoid-answering system to thoroughly scan all related documents in the information resources by changing the number of responses to be returned as answer (Harabagiu et al., 2003).

2.2.2 Non-Factoid Question Answering

Unlike FBQA, NFQA have an unlimited variety of syntactic forms without an explicit connection between their syntax and expected answers. This classification includes:

- **Definition Questions**: Usually start with the question word "What" and "Who" such as *"What is the Nobel Prize?"* or *"Who is Colin Powell?"* Voorhees (2001) suggested that "*it is an important type as it occurs relatively frequently in logs of web search engines*". Responses for definition questions emphasize nugget recalls rather than exact answers. In this context, systems are expected to return a summarized sentence or a short paragraph about a particular person or thing. For example, a correct answer for the previous question would imply important events in Colin Powell's life (birth, graduation and marriage), his major positions and achievements and any other interesting information. This type of question was introduced for the first time in TREC 2003. Systems generally used more complex techniques than those used for FBQA. Mostly they first retrieved passages about target using recall-oriented search then performed several types of text understanding, summarization and reasoning processes (Voorhees, 2003a). Furthermore, the evaluation of systems answering definition questions is much more difficult than the evaluation of systems tackling FBQA due to uselessness of right and wrong judgments used to evaluate FBQA responses.

- **Analytical Questions:** For this type of questions one cannot generally anticipate what might constitute the answer as in "*what has been Russia's reaction to the U.S. bombing of Kosovo?"* (Small et al., 2004). Moreover, in many cases the answers to such questions are not explicitly mentioned in the knowledge resources. Therefore, answering these questions entails conducting a clarification dialogue with the user in order to have a semantic interpretation of questions and candidate answers as well as to have a comprehensive and deep inferential analysis of the knowledge elements of knowledge resource. This type of questions is similar to the complex interactive QA task introduced in TREC 2006.
- **Reasoning & Explanation Questions:** The most prominent questions in this type are "*why*" and "*how to*". For example, "*Why does ice float on water?*" and "*How to enable command auto complete by searching history in windows*". Finding answers to such questions involves searching for argument relations in texts such as (Causal, Motivation and Purpose). Relatively, there are few systems presented with the aim of handling reasoning and explanation questions; the systems were restricted to specific domains with several limitations. This type of question has not been addressed in TREC annual conferences. Section 2.5 addresses this type in more detail as it is our main concern in this work.

2.3 Question Answering Approaches

As we discussed in the previous sections, there are many systems that have been implemented to automatically answer questions. However, developing and implementing a QA system is not an easy task. Inspired by the QA systems presented in Section 2.1.2, we have developed Figure 2-4 that illustrates the generic architecture of a typical question answering system. It comprises three main components: *Question Processing* module, *Passage Retrieval* module and *Answer Processing* module. Each of which can be sub-divided into lower level operations.

Throughout the following subsection we briefly review some of the existing approaches that have been reported in the literature for the three modules mentioned in Figure 2-4, taking into consideration the well performed systems in TREC.

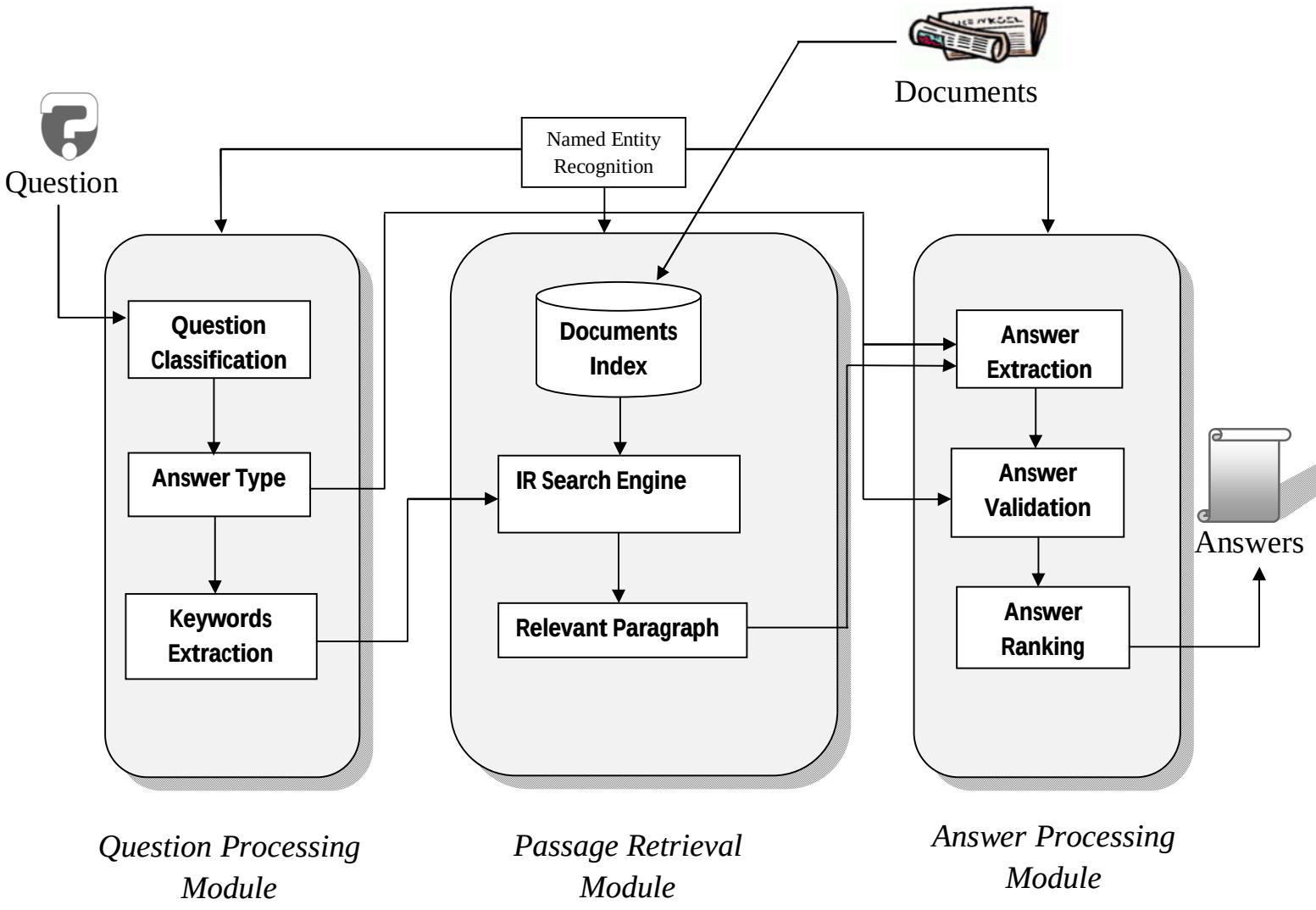

Figure 2-4: Generic architecture of Question Answering Systems.

2.3.1 Question Processing

The first component in any QA systems is question processing. The aim of this module is to parse the input data presented as a natural language question in order to understand the posed question. The output of this module should be representations of the question in multiple forms (semantic, logical, Boolean etc.). These representations are used by the rest of the system's components to extract the correct answer. Given a question expressed in a natural language this module includes:

- **Question Classification:** In this stage the system is finding the type of the question being asked (e.g., Where, When, Who, Why) using the taxonomy of questions built into the system. In some cases a further classification is needed to better identify the question type. For example, in PowerAnswer4 system a language model was built for each class using questions from previous TRECs to automatically create question classes. The model was developed based on a set of features which involved: *stemmed keywords, morphological alternations* for each of the keywords and *named entity tags*. These features were extracted from the answers judged as correct for each of the question classes (Moldovan et al., 2007).
- **Answer-Type Identification:** Here the system is inferring what kind of answers is expected (e.g., Location, Proper Person, Organization etc.) to help the *Answer Processing* module retrieve the correct answers. The simplest categorization is performed by checking the interrogative word introducing the question. For example, it is obvious that the answer type is PERSON for the question "*Who invented the toothbrush*". The Named Entity (NE) concept that was first defined in MUC (Gaizauskas and Wilks, 1997) plays an important role in determining the answer type of the Factoid Questions. In contrast, this technique does not apply for NFQA (Pasca and Harabagiu, 2001). Certain types of inquiry words that belong to FBQA include different kinds of answers. For example, the word "*what*" does not tell us about the information asked by users; we notice that the following question "*What is the biggest city in the World*" intends Location information, whereas the question "*what is the first month of the Hijri calendar*" asks for a Date. To solve such problem the LLC system included a concept named Focus: *"a word or a sequence of words which define the question and disambiguate the question by indicating what the question is looking for"* (Harabagiu et al., 2003). For example, *"What is the largest city in Germany?"* the focus is *largest city* (Moldovan et al., 2000). Instead of knowledge-based analysis techniques, the TextRoller system employed predefined indicative patterns of textual expressions in order to find the answer type, for example the pattern "city name; comma; country name" indicates answers for "*Where*" questions. Accordingly, the presence of the string "Milan, Italy" in any text can be considered as an answer for the following question *"Where is Milan?"*

(Soubbotin, 2001). As the questions presented in TREC were becoming more challenging over the years, more complex approaches were explored. The system that got the highest scores is the one that constructed a hierarchy of answer types and then induced a classifier which assigned a type for each question based on Machine learning approaches (naïve Bayes, decision trees and support vector machines). The PowerAnswer3 system that was ranked first in TREC 2006 used a hybrid approach including heuristics and machine learning algorithms in order to disambiguate inquiry words and predict the answer types. A maximum-entropy model was constructed which incorporated a set of features including a variety of attributes such as: POS, lemma, head information, parse path to question word, named entity information, and set-to-set lexical chains derived from eXtended WordNet which links the set of question keywords to the set of potential answer type nodes. The maximum-entropy model performed well in answer type detection with an Error rate of 11% (Moldovan et al., 2006).

- **Formulating a Query:** This process involves converting the original question to a query by determining the list of keywords to be used by a search engine in the *Passage Retrieval* Module. The common approach is the bag-of-words (BOW) model where questions are represented as an unordered collection of words, disregarding grammatical structures. For example, the question *"who invented the paper clip?"* is converted to: [paper ∧ clip ∧ invented]. It is very often that stop words, punctuation and the focus of questions are removed as their role is just to form the context of questions; furthermore all the inquiry words are stemmed to remove morphological variations associated with documents words. In (Moldovan et al., 2000) a set of ordered heuristics were used, each of which added a set of keywords, for example, *Heuristic 1:* adds quoted expressions, *Heuristic 2:* adds all named entities recognized as proper nouns, *Heuristic 3:* adds complex nominals and their adjectival modifiers and *Heuristic 4:* adds all other complex nominals. One further step that many systems make is the expansion of queries so that correct answers do not be missed. Most systems use the knowledge base in WordNet to add more keywords to queries. The system presented by Harabagiu et al. (2001) used three different sets of keyword alternations based on the following three heuristics that decided which word and form of alternations is to

be used: 1- ***Morphological Alternations:*** in case that no answer was found, question words determine the keyword to be altered. For the question *"who invented the paper clip?"* all inflections of the verb invent will be added so that the expanded query would be as follows [paper ∧ clip ∧ (invented ∨ inventor ∨ invent ∨ invents)]. 2- ***Lexical Alternations:*** here the system should exploit WordNet resources to enhance the recall of answers by adding semantically related terms. For example, one synonym of the verb *invents* which is *devise* and this could be included into the query.

3- ***Semantic Alternations and paraphrases:*** this set of alternations applies to collocations form if *"(a) they are not members of any WordNet synsets containing the original keyword; (b) have a chain of WordNet relations or bigram relations that connect it to the original keyword*" (Harabagiu et al., 2001). For example, in the question *"Where do lobsters like to live?"* the verb *prefer* can be added to the expansion query since it is a hypernym to the verb *like* [lobsters ∧ (like ∨ prefer) ∧ live].

Different techniques were employed by PALANTER system (Harabagiu et al., 2005) to select keywords from complex questions in the relationship task introduced in TREC 2005 (Voorhees, 2005). The system heuristically assigned a weight to each keyword extracted based on the approximation of keywords' importance to queries *"the highest weights were assigned to proper names, followed by comparative and superlative adjective, ordinal numbers, and quoted text"*, then query expansion was performed by adding synonyms and keywords alternations from a database of similar terms. However, generating expanded queries has its own problems as it generates complicated queries. Moreover, the size of indexes to be matched against the document collections requires more computational processing. Bilotti (2004) reported that the increase in the number of retrieved documents when using morphological expansion comes at the expense of moving relevant documents further down the ranking list.

2.3.2 Passage Retrieval

Passage Retrieval (PR) is the core module of QA systems which is responsible for reducing the search space determining the quantity of texts to be passed to the final module in the system architecture. As the syntactic and semantic parsing of whole collection is a time consuming process, in some cases before the queries-answers matching begins the documents in collections are transformed into other representations so that more efficient search can be performed.

Although numerous strategies are involved in this component, BOW is the standard approach for finding passages in document collections. Many studies pointed out that combining structural information with BOW would improve the accuracy. Quarteroni et al. (2007) handled definition questions by employing predicate-argument structures (PAS). The results showed that incorporating PAS into BOW gave slight improvement with F-score of 70.7% compared to BOW alone which got an F-score of 69.3%.

Another research presented by Surdeanu et al. (2008) considered the problem of extracting "*how-to"* questions using a large community-generated collection from Yahoo! answers logs. Surdeanu et al. (2008) explored a set of different features (similarity, translation, density and frequency) and concluded that "*syntactic dependency parsing and coarse semantic disambiguation yield a small, yet statistically significant performance increase on top of the traditional bag-of-words representation*".

Also the work developed by Verberne et al., (2010) to handle *why*-QA studied the inclusion of structural information on cue phrases, noun phrases, question focus and the syntactic structure of questions. They investigated different features sets based on structural overlap between questions and answers: syntactic structure of questions, semantic structure of questions, synonyms, WordNet relatedness and Cue words. They found a significant improvement in terms of MRR (from 0.249 to 0.341).

Boolean indexing (implementing the operators AND, OR and NOT) is another approach that has been suggested by a number of studies (Saggion et al., 2004; Moldovan et al., 2000). This technique requires less processing time; however, the number of documents returned by this approach may be large and unordered since it does not have any built-in way of ranking the

matched documents. Therefore, a special consideration has to be given for filtering and ordering the generated list of documents.

Moldovan et al., (2000) used radix sort to perform paragraph ordering based on the notion of *paragraph-window* by introducing three different scores: the largest *Same_word_sequence_score*, the largest *Distance_score* and the smallest *Missing _keyword_score*.

Statistical density-based information about term occurrences in passages has also been investigated for building PR module. IBM group (Ittycheriah and Roukos, 2002), one of the top 5 TREC 2002 contestants, incorporated a web search feature by using a supervised corpus of questions and answers in order to extract 5-gram lexical answer patterns occurring in the answer.

2.3.3 Answer Processing

The final stage in a QA system and the most important one is the *Answer Processing* Module (AP). The role of AP module is to extract a list of candidate concise answers from the relevant passages. The Named Entity Recognition unit as shown in Figure 2-4 plays an essential role in validating FBQA answers. Unless it has been performed in advance and stored along with the retrieved documents, several processing steps which include tokenization, POS tagging and sentence splitting, may be performed in this component depending on the approach that has been applied. Semantic parsing is needed for this stage to extract answers for NFQA.

Murata et al. (2000) calculated sentential similarity between a question and each sentence in the target texts according to POS, syntactic and NE information. They suggested that this action would improve the accuracy of the retrieved answers since it is searching for consistency between NEs in target collections and questions.

The prominent problem for this method is the high computational costs as it treats all of possible expressions in documents equally. However, Mori et al. (2003) employed the A^* search algorithm as a way to control searching which is in turn reduces the calculation cost. The algorithm processes the most promising candidates first and delays the processing of the others.

One simplistic approach is the one that employed surface text patterns (Soubbotin et al., 2001; Cooper and Ruger, 2000; Ravichandran and Hovy, 2002). For the TextRoller system which obtained the best score in TREC 2001, the designers suggested checking "*the answer candidates for the presence of certain predefined patterns to which scores were assigned beforehand, i.e. independently of the question text analysis. Candidate snippets containing the highest-scored patterns are chosen as final answers"* (Soubbotin et al., 2001).

Inspired by the good performance obtained by the TextRoller system, Ravichandran and Hovy (2002) used machine learning of bootstrapping to build a large tagged corpus so that they can automatically learn such patterns starting with a few examples of QA pairs along with their precision. For instance, for BIRTHDATE questions like "When was X born?" they selected pairs of question and answer terms such as Gandhi 1869, Newton 1642, Mozart 1756, etc. Then they submitted these pairs to a search engine and downloaded the top 1000 web documents for each pair. Next they passed each document into sentence breaker and tokenizer to extract phrases that contain both the question and the answer terms. This procedure produced a set of patterns as those included in Figure 2-5. For the extracting answers stage their algorithm replaced question terms in each sentence by question tags ("<NAME>") and then searched for the presence of each pattern and selected the words matching the tag "<ANSWER>" as a candidate answer and finally it sorted these answers by their pattern's precision scores (Ravichandran and Hovy, 2002).

A similar approach was employed by Greenwood and Saggion (2004) to answer factoid and list questions along with one type of NFBQ, definition question, using a library of patterns identified by corpus analysis. A more complex approach incorporated the semantic type extraction. This approach requires a system to recognize all entities of the expected answer type (Greenwood, 2004).

<NAME> (<ANSWER> -)
<NAME> was born on <ANSWER> ,
<NAME> was born in <ANSWER> ,
<NAME> was born <ANSWER>
<ANSWER> <NAME> was born
- <NAME> (<ANSWER>
<NAME> (<ANSWER> -

Figure 2-5: A pattern list example extracted by Ravichandran and Hovy (2002).

Answer ranking is a solid challenge for any question answering system. To improve this task, Greenwood used the frequency of candidate answers occurrences within the retrieved documents in addition to the overlap between questions and sentences in which answers may be found (Greenwood, 2004). PowerAnswer2 system exploited answers redundancy in large corpora, Internet and Wikipedia, where the most redundant answer was added to the keyword features leading to another ranking of the answers produced by the AP module (Harabagiu et al., 2005).

2.4 Arabic Question Answering Systems

In the last few decades many QA systems have been implemented and presented at international conferences for instance TREC and CLEF. Those systems have been built mainly to support users of the English language, many western languages such as: German, French, Dutch, Portuguese etc., and some Asian languages such as Japanese. But very few systems have been developed for Arabic, though it is a more common language than many of the others. The main concern of the Arabic QA systems was extracting answers for FBQA. To our knowledge, NBQA such as "*why*" and "*how to*" questions have not been investigated before.

One of the first known systems oriented to Arabic language is AQAS system (Mohammed et al., 1993) that handled propositional interrogative and argument interrogative sentences. In their work they created several linked frames to represent their knowledge base of radiation diseases. Each frame included specific information (size, shape, effect, contents etc.) which represented a particular situation of the domain. The parser converted each query into tree structure that reflected the required part (the thing we ask about) and known part (what we need to know) by applying dictionary checking and morphology processing, the interpreter component then used this representation to decide which question module is to be activated. The system also accepted a user's declarative statement to enhance the existed knowledge base. There is no information about the efficiency of their system as neither results nor evaluations have been presented.

A more standard system addressed Arabic Factoid question, is the QARAB system (Hammou et al., 2002) that is composed of three basic modules (question analyzer, information retrieval and passage selection). The system processed input questions using shallow language

understanding without performing any semantic analysis. It then returned a short passage representing the answer over a collection of documents extracted from Al-Raya newspaper. The system was constructed using a relational database management system (RDBMS) consisting of a set of tables. The tables contained rows of roots, stems, weights, occurrences and locations of all words extracted from the entire document collection, as well as tables that stored information about paragraph and documents (date, title and path). Several NLP tools were used in order to build their own Arabic lexicon and to process queries. These tools included: *tokenizer, POS tagger, word's feature finder (gender, number, person and tense), stop words remover and proper noun phrase parser.*

The system employed a query expansion technique and BOW model to retrieve a ranked list of candidate documents. Furthermore, question classification was based on the interrogative particles that precede the question. For the purpose of evaluation, 113 questions were presented to four native Arabic speakers to judge the correctness of the answers. The system obtained a recall of 97%. However, the results are surprising compared to other scores achieved for the English language so its reliability may be low as Benajiba et al (2007a) stated "*There are no Arabic QA tasks which provide a test-bed allowing a general test for any Arabic QA system"*.

ArabiQA is an Arabic QA prototype which was also developed by Benajiba et al. (2007a) to handle Arabic Factoid questions. The authors implemented each component, tested it and evaluated it separately. They focused on Named Entities Recognition (NER) module as it is needed for most of the system's components; the module based on Maximum Entropy (ME) approach as they believe that *"this approach tackles the problem better than others because of its features-based model"* (Benajiba et al., 2007a). For implementing the *Passage Retrieval* component, they adapted JIRS system for Arabic language (Benajiba et al., 2007b). JIRS system first used an n-gram model to index documents (Soriano et al., 2005). During the retrieving process it assigns a weight to each document depending on the terms' relevance between questions and passages. Then it selects the top *(m)* relevant passages to extract n-grams from each one. Finally it employs the Density Distance Model to compare n-grams for both queries and passages, where the passages that have smaller distance among question structures are supposed to get more weight. Authors tested the performance of their JIRS adapted system over a collection consisting of 11,000 documents of Arabic Wikipedia, 200

questions and a list containing all possible answers. They reached a coverage (ratio of the number of the correct retrieved passages to the number of the passages returned for a question) of up to 59% and a redundancy (average of the number of the passages returned for a question) of 1.65 without performing any text preprocessing, when a light-stemming was applied a coverage raised up to 69% and redundancy up to 3.28 (Benajiba et al., 2007b)

Another attempt towards an Arabic FBQA system was presented by Kanaan et al. (2009). The system returns as output a set of ranked documents with texts containing the answers. NLP tools were used to construct a lexicon comprising information on the morphology, phonology, syntactic argument structure and semantics of words. The system is closely similar to QARAB system (Hammou et al., 2002) in terms of adopting RDBMS for implementing their IR unit, where several tables were created to contain entries for sorted information related to *Words, Query Weight, Similarity of the Query, Extracted Roots and Term Weighting.* The IR unit was implemented using Salton's Vector Space Model in order to calculate the degree of similarity between documents and targeted queries. For evaluation, they used interpolating procedure based on *recall* (the fraction of the relevant documents that have been retrieved) and *precision* (the fraction of the retrieved documents that are relevant) measures. 12 questions were tested over a collection consisted of 25 documents gathered from the Internet in addition to some relevant documents manually selected. The authors claimed to get results that are close to the reported performance of the traditional Vector Space Model. Unfortunately, the results were not clearly presented as the results figure was missing from the paper.

Recently, Akour et al. (2011) used the same methodology presented in (Hammou et al., 2002; Kanaan et al., 2009) to introduce the QArabPro system for FBQA and NBQA based on a set of separate rules for each type of questions. The test was conducted over a collection of reading comprehension texts collected from WIKIPEDIA and they obtained an overall accuracy of 84% which is also a very surprisingly high result compared to the others obtained for the English language. Furthermore, the authors used the same method to handle all question's types including "why" questions; they reported that *"the system relies on shallow language understanding and do not attempt to understand the content at the semantic level"* (Akour et al., 2011).

However, many studies suggested that successful techniques for FBQA have been demonstrated to be not suitable for questions that expect explanatory answers since knowledge about discourse relation is crucial to answer this type of question (Kupice, 1999; Breck et al., 2000; Verberne et al., 2007). For example, in their work they marked the word "حيث" as stop word that has to be omitted from a query/document processing. In fact, this word is used in today Arabic language to indicate *Causal* relations that lead to answer "*why*" questions. Moreover, the authors claimed that they handled the question type "كيف" "*how to*". However, what they actually handled is the type (how much/many) "كم" which is totally different from "how to".

2.5 Related Work

As discussed above, NFQA is much less addressed by researchers in the field of QA systems than FBQA due to the linguistic knowledge required for approaching such questions. However, in recent years more and more researchers have become interested in adapting new methods that would be able to handle explanation and reasoning questions.

2.5.1 Relation Extraction

Many studies conclude that the wise exploitation of discourse structure (i.e. understanding the role of each sentence in the text and how they are related to each other) can improve the effectiveness of extracting answers for NFQA (Kupice, 1999; Breck et al., 2000). Therefore, several studies have been presented for mining semantic relations. These studies have mostly focused on the detection and extraction of the *Causal* relation since it is a fundamental relation in many disciplines including QA. Furthermore, it closely relates to some relations (TEMPORAL and INFLUENCE) and can be seen as a supertype of a number of relations such as (CONDITION, CONSEQUENCE and REASON) (Blancol et al., 2008).

The early attempts for detection causation in written texts made use of hand-coded and domain-specific knowledge bases. For example, in the COATIS system (Garcia, 1997) a model was built for casual knowledge acquisition by locating *Causal* relations between two expressions of actions in French texts. The model was created by doing manual classification of indicator verbs in technical domain. It applied the strategy of Contextual Exploration which decides if the located indicator is likely to express a *Causal* relation as well as to identify the

argument of relations. In order to confirm the presence of a *Causal* relation in a sentence, the system took into account the context in which the located indicators appear. This involves considering relevant information in texts such as morphologic and morpho-syntactic (the occurrence of an infinitive verb preceding or following the indicator). The author reported to reach a precision rate of 85%.

Another attempt presented by Khoo et al., (2000), in which English linguistic patterns were identified to extract cause-effect templates that are explicitly expressed within sentences from medical abstracts. They developed a parser to convert sentences and the causality patterns into conceptual graphs which reflect the syntactic structure of the target. The graphs representing the patterns were then matched against the graphs representing the sentences to locate the presence of *Causal* relations and to fill the cause-effect template with the textual parts that match each slot. They obtained accuracy of 0.41, 0.48 for extracting the *cause* and the *effect* slots respectively.

A semi-automatic approach was proposed by Girju and Moldovan (2002) to identify *Casual* relations and used lexico-syntactic patterns. It was called semi-automatic since the patterns were extracted automatically whereas the process of pattern ranking and validating was performed manually. The authors concentrated their work on the pattern *<NP1 verb NP2>* reporting that it is the most frequent intra-sentential pattern that indicates causation. Their approach used WordNet as the main knowledge resource from which pairs of noun phrases was extracted. A list of verb expressions was then constructed by searching a number of document collections for each pair extracted from WordNet. Finally several semantic constraints were imposed on *NP1, NP2 and verbs* for ranking the patterns and validating that the verbs from the list were relating to the context. Constraints comprised observations and statistics derived from WordNet. Testing was conducted using (TREC-9 2000) collection of texts; two human subjects were asked to judge whether the relations returned by the system are *Causal* ones, the average accuracy obtained was 65.6%.

Machine learning techniques were employed by a number of studies for automatically harvesting causal patterns. An example of these studies is the one presented by Blancol et al. (2008) in which the authors concentrated their work on the syntactic pattern *[VP rel C], [rel C, VP]* when performing pattern classification; they state that this pattern

comprises more than half of the causations found in TREC5 corpus. Where the *C* symbol in the pattern stands for Causation, *VP* for a verb phrase and *rel* for a relator (preposition or conjunction) that was restricted to the occurrences of one of the following words (after, as, because and since). For pattern validation, an algorithm was trained to learn to discriminate whether or not a pattern referred to causation using a set of lexical, syntactic and semantic features extracted mainly from WordNet. For example, [***Relator*** *(A relator can encode a causation always or sometimes),* ***Relator left and right Modifiers*** *(adverb + after almost always signals a temporal relation, not a causation. as + preposition can hardly signal a causation),* ***Semantic Class Cause Verb*** *(if the relator is after and the cause verb semantic class is be-v-3, then it is a temporal relation not a causation),* ***Verb Tense Cause and Effect Verb*** *(if the relator is* ***"as"*** *and the effect verb is conditional, then is not a causation. If the effect verb is passive, then it is more likely to express causation)*]. Conducting the testing phase, the system obtained a *recall* of 0.84 and *precision* of 0.95 for cause cases; and a *recall* of 0.86 and *precision* of 0.96 for not cause cases. However, the authors pinpointed that "*the model is only able to classify correctly the causations signalled by the relators* ***because*** *and* ***since***".

More recently, a less supervised algorithm was proposed by Itto and Bouma (2011) by exploiting Wikipedia as a raw knowledge base. In the pattern acquisition phase, all sentences extracted from Wikipedia are converted into lexico-syntactic patterns each of which represents a pair of events connected by a semantic relation. In the causal pattern extraction phase, a supervised algorithm decides which of these patterns encode causality. The pairs of events denoting *Causal* relations are then used to learn new patterns. The reliability of each pattern is calculated and the most reliable patterns are kept. The acquired patterns were applied to specialized documents collected from customer service responses on medical equipment in order to evaluate their efficiency. With this approach the researchers achieved high scores with *precision* of 76.5% and *recall* of 82%.

2.5.2 Why and How to Questions

Since finding answers to "*why"* and "*how to"* questions has been considered as a challenging task, few studies have been dedicated by the QA community to deal with such task. Suzan Verberne intensively worked on finding answers to "*why"* questions by approaching the

answer extraction problem as a discourse analysis task. In (Verberne et al., 2007) Rhetorical Structure Theory (RST) was adopted for discovering discourse structure. In their work, Verberne and her colleagues used RST Discourse Treebank created by Carlson et al. (2001). This Treebank has been manually annotated with discourse relations proposed by Mann and Thompson (1988) in the framework named RST. Verberne selected from the Treebank a number of rhetorical relations that indicate arguments in texts, and in turn constitute candidates answers for why questions. To evaluate their work, they selected seven RST-annotated texts and asked English native speakers to read each text and formulate questions that were supported by the source text. Subjects were also asked to identify the answers for each of their questions. The system was able to return a correct answer for 58% of the questions collection.

Verberne (2007) shifted the *why* QA task towards paragraph retrieval rather than a textual span stating that 61% of the answers are exactly one paragraph long. Furthermore, she mentioned that *"in realistic applications of why-QA using RST, the system has to deal with automatically annotated data, consequently, performance must be expected to decline with the use of automatically created annotations"*.

Recently, Verberne investigated different supervised learning algorithms (genetic algorithms, logistic regression and SVM) in order to find the optimal ranking function that is used for re-ordering the set of candidate answers (Verberne et al., 2009). She employed a set of features extracted from questions and candidate answers retrieved by a search engine. Most of the features were linguistic ones (syntactic, WordNet, Cue word etc.) and their values reflect the similarity between questions constituents and answer items. Experiments showed that logistic regression was the best learning technique with MRR of 0.34.

Parsed and Josh (2008) tried to find out to what extent discovering *Causal* relations in texts would cover "*why*" questions. They made use of the annotated Penn Discourse TreeBank (PDTB) corpus as a resource of discourse relations. This corpus contains annotations of explicit and implicit discourse relations holding between two abstract objects in texts such as events, facts and propositions. They selected QA pairs related to three texts from the data collection developed by Verberne et al. (2007) which is also subset of the PDTB corpus. The

results obtained showed that 71% of the collected questions were correlated with one of the *Causal* relations.

Some efforts were conducted to build why-QA systems directed to the Japanese language (Fukumoto, 2007; Mori et al., 2007; Shima and Mitamura, 2007; Higashinaa and Isozaki, 2008). The earlier systems (Fukumoto, 2007; Mori et al., 2007) heavily depended on hand crafted linguistic patterns that were matched against targeted documents in order to extract an appropriate string as an answer candidate. The recent systems focused on using heuristics and machine learning-based approaches (Shima and Mitamura, 2007; Higashinaka and Isozaki, 2008).

Fukumoto (2007) created his system to handle three types of questions (why, how and definition). For "*why"* questions, a number of clue words that might be included in question sentences along with extraction and non-extraction patterns have been set to locate the reason part of a causal sentence. The system was tested over 100 questions belonging to the three abovementioned types; it returned correct answers to 30 questions. The author reported that it is important to add more patterns to the list as a way to improve his system.

Similarly, the system implemented by Mori et al. (2007) constructed its lexico-syntactic patterns for different types (definitional, why, how and factoid) by adopting two measures ***Appropriateness of writing style*** *(how appropriate is the writing style of the candidate in terms of the given question)* and ***Relevance to the question*** *(how relevant is the candidate to the topic of the question)*. The system achieved better performance for definition-type question than other types. The authors justify this because the question classifier was performed poorly as many of non-factoid questions are incorrectly classified into the type of factoid.

The last (third) version of the JAVELIN system that was originally implemented for factoid English language has been extended to accept non-factoid question including "*why"* type and "*how"* type questions for the Japanese language (Shima and Mitamura, 2007). In its third edition the system used an annotated database with various information such as morpheme text chunks, POS and named entities along with predicate-argument analysis. The adoption of machine learning technique was incorporated with hand crafted cue words that may identify the type of relation sentences. The results obtained from the system showed that the

performance was less efficient than the versions created for factoid questions. One reason for that is the small number of the examples available for the training phase (30 questions).

Another system that made use of machine learning is presented by Higashinaka and Isozaki (2008) with the aim of ranking a given set of candidate answers for Japanese why-questions. The study based on the assumption that answers are of a one sentence or paragraph long and to be extracted from top-N documents returned by a document retrieval module. The features (causal expressions, causal relation and content similarity) were mainly based on causal expressions extracted from semantically tagged corpora. The answer candidate ranker obtained MRR of 0.305 for top-5.

The system developed by Surdeanu et al., (2008) took advantage of the abundant content provided by one of the social websites[3] to rank a set of answers for "*how to*" questions in English language. The corpus was created upon U.S.Yahoo! Answers logs by excluding the questions that do not have any answer among the best ranked answers and keeping only the questions and answers that contain at least 4 words each. In doing so, the corpus had about 142,000 question-answer pairs. Three different types of machine learning methodologies - unsupervised learning, discriminative learning and class-conditional learning - were used for the main components of the system, respectively answer retrieval, answer ranking and question to answer. Moreover the features have been classified into four groups in order to measure the similarity between questions and answers, keyword density and frequency, the correlation between each question answer pairs and to encode questions into answers transformations. The authors selected as a baseline the output of the answer retrieval model that precedes the answer ranking model; the system achieved a 14% improvement in MRR at N=15 over their baseline.

2.6 Summary

Different methods and approaches of using NLP techniques in QA systems have been explored in this chapter. For each of the QA systems components of *Question Processing*, *Passage Retrieval* and *Answer Processing*, key research problems have been identified. This was followed by a survey of QA systems implemented for the Arabic language; to the best of

[3] Yahoo! Answers

our knowledge, no previous systems has been developed to deal with "*why*" and "*how to*" questions in the Arabic language.

The chapter also reviewed QA systems presented to handle Non-Factoid questions with the focus on the systems targeting "*why*" and "*how to*" questions. Among existing NBQA systems, those which utilize reasoning capabilities and linguistic information have been shown to achieve greater performance in English and Japanese languages. In this context, exploiting texts structure plays an essential role when approaching non-factoid questions. As such, our approach for answering "*why*" and "*how to*" questions rely on discovering causation and explanation in Arabic texts.

In the next chapter, we will investigate Arabic literature to build the first model of our QA system i.e. *Pattern Recognizer* model. This model is accountable for the mining of causation and explanation within sentences.

Chapter 3

Pattern Recognition

3.1 Introduction

The goal of this research is the automatic detection of *Causal* and *Explanatory* relations expressed in Arabic texts which can lead to answer "*why*" and "*how to*" questions. This chapter describes the first step in pursuing this aim i.e. indicating the presences of *Causal* and *Explanatory* relations within sentences. To fulfil this goal, a *Pattern Recognizer* model has been developed to signal the presence of cause-effect/method-effect information within sentences. The approach adopted in this study makes use of a set of hand-crafted linguistic patterns indicating the presence of the targeted relations defined by the researchers.

A number of studies made for other languages have used machine learning approaches in order to automatically construct syntactic patterns that may encode causation. However, these studies have exploited the electronic knowledge resources which are available for the language they addressed. These resources have facilitated the development of robust machine learning models. For example, *large annotated corpora, WordNet, dictionaries, Wikipedia etc.* Furthermore, such studies have restricted their work to the extraction of one kind of lexico-syntactic patterns such as <*NP1 verb NP2*>.

Unfortunately the Arabic language, so far, lacks mature knowledge base resources upon which machine learning algorithms heavily rely. Recently, Leeds Arabic Discourse Treebank (LADTB)[4] has been presented as an Arabic corpus annotated with discourse relations. This corpus contains approximately 500 *Causal* relations; however, the syntactical patterns of the Arabic relations are relatively large compared to the size of the available training corpus. Thus, 500 relations are insufficient instances for systems designed to learn and train features

[4] www.arabicdiscourse.net

involving statistical component, resulting in a poor learning performance. On the other hand, the restriction to one type of syntactic patterns is limited in scope and unable to reveal the richness of the Arabic texture.

In this work, we use the expressions *Causal* and *Explanatory* as super-class terms where each refers to a number of relations that belong to the same category. In this context, when the term *Causal* is used, we refer to the relations (Causal, Result and Purpose). In the same way, the term *Explanatory* is used to refer to (Explanation, Interpretation and Evidence) relations.

3.2 Causation and Explanation

Causation and explanation are two textual relationships that relate two situations. The *Causal* relation occurs between an event (the cause) and a second event (the effect) where the second event is understood as a consequence of the first. On the other hand, the *Explanatory* relation is presumed to happen when the second event presents an explanation for the situation stated in the previous one.

Few studies have touched on the topic of defining and distinguishing causation in Arabic texts Haskkour (1990). These studies have referred to causation broadly in the course of their research while discussing other language phenomena (Ibn Jinni, 1952; Abu-Hilal Al-Askri, 1952; Al-Zubaydi, 1888).

On the other hand, no work, to our knowledge, has been devoted to the study of explanation in Arabic. However, locating *Explanatory* relations are crucial step in the process of finding answers to "*how to*" questions. In this research, Arabic texts have been analyzed to observe the behaviour of such relation.

3.2.1 Expression of Causation in Arabic Text

Haskkour (1990) has extensively surveyed *Causal* relation in the written Arabic literature. She has argued that causation from the perspective of grammarians can be classified into two main categories. The first one is: ***السببية بالملفوظ (verbal causality)*** which can be captured by the presence of nominal clauses e.g. [المفعول لأجله (Accusatives of purpose), المفعول المطلق (Cognate accusative)] or by causality connectors such as [لذا (therefore), بسبب (because), من اجل (for)] even though these connectors may in many cases signal different relations other

than causation. The second category is: ***السببية بالملحوظ*** (***context- based causality)*** that can be inferred by the reader using general knowledge without locating any of the previous indicators. This category includes various Arabic stylistic structures that express causality implicitly such as [الاستئناف(resumption) – الشرط (condition) – الاستثناء (exception)].

Generally speaking Haskkour (1990) observations can be summarized in a similar way to that presented by Blancol et al. (2008) who made the following distinctions for *Causal* relations.

- **Marked or Unmarked**: In case that a *Causal* relation is indicated by a specific linguistic unit it is a ***marked relation,*** for example, ***"****The flight has been cancelled due to a volcano eruption****".*** The other case is ***unmarked relation,*** for example, *"Be careful. It's unstable"*.

- **Ambiguous or Unambiguous**: ***Unambiguous*** connectors are those which always indicate *Causal* relations in text like "because, due to". On the other, hand they are considered ***ambiguous*** if they are associated with multiple relations. For example, the connector "حتى" may in some cases expresses causation in the sense of "because, since, as" whilst in other cases it refers to the *Temporal* relation indicating motion towards and at the same time arrival at an object; this behaviour is illustrated in sentence (3). It also exercises like other copulative particles in the sense of *"even"* where no independent influence upon the following noun, but rather remains under the same government of the preceding noun. Consider for example the occurrence of "حتى" in sentence (4); in which the following noun "the teachers" receives the same action as the preceding noun "the head of school" i.e. arriving to the meeting.

(3) نام الطفل البارحة حتى الصباح.

"The baby slept last night ***till*** morning"

(4) وصل المدير الى الاجتماع حتى المدرسون.

"The head of school has arrived to the meeting ***even*** the teachers"

- **Explicit or Implicit**: In the ***explicit*** relations, both arguments (cause, effect) are present; on the other side a relation is considered as ***implicit*** if any of its elements is missing. ***Implicit*** relations are frequently used in rhetorical expressions especially in

novels, poetry and the holly Quran. Consider for example the following sentence *"we said: strike the stone with your stick, and there gushed forth form it twelve springs"*[5] In this sentence, the action of striking the stone which was the result of the appearance of water –*he stroked, so it exploded*- is not stated explicitly.

3.2.2 Identifying Causal and Explanatory relations

The definition of implicit relations in Arabic has been controversial among linguists and raised many interpretations and acceptance issues. It is not the aim of this study to add to these controversies but we will restrict our study to the extraction of explicit relations indicated by ambiguous/unambiguous markers.

Altenberg's *typology of causal linkage* (Altenberg, 1984) which covers linking words and describes which clause or phrase is the *cause* and which one is the *effect* was of great importance for extracting *Causal* relations in English. Unfortunately, such a list does not exist in the Arabic language neither for causation nor for explanation.

Discourse connectives such as "لذلك، عن طريق، لكي" have an important linking function that link two clauses together. Traditional Arabic grammarians have considered these items to be function words and they have referred to them by the term "ادوات" which means 'tools' or 'devices'. In their study, Arabic grammarians have provided comprehensive descriptions of these linguistic devices classifying them as a grammatical class whose members operate within sentence boundaries (Kammensjo 2010; Hatim 1998).

In order to locate the elements that signal causation and explanation in Arabic texts, we have surveyed all causative connectors from the perspective of grammarians mentioned in (Haskkour, 1990) and the verbs that are synonymous with the verb "يسبب" (cause) such as "يؤدي، ينتج، يفضي...". Likewise, we have studied the grammatical particles presented in ***Mughi al-labib*** (Haskkour, 2009) that indicate causation. We have also investigated Arabic discourse in order to find out the items that are commonly used in modern Arabic texts to indicate causation and explanation such as "من خلال ، حيث".

[5] The Holly Quran 2:60

3.3 Constructing the Linguistic Patterns

The method adopted in this study is similar to the pattern-matching and slot-filling in Information Extraction (IE). It applies a set of pre-defined linguistic patterns to a natural language text in order to match particular type of a relation and extract cause-effect/method-effect information. The patterns have been generated by analyzing a data collection extracted from a large untagged Arabic corpus called ***arabiCorpus***[6].

This corpus contains non-vocalized texts and thus it is representative of real-world Arabic texts; furthermore it is available online for exploration. The corpus consists of a variety of resources classified into five main categories (Newspapers, Modern Literature, Nonfiction, Egyptian Colloquial and Premodern). It also provides useful searching tools that help studying lexical items and their syntactical categories in the sentences in which the link words under scrutiny appear.

Furthermore, it has a number of filters that allow the searching of specific word included or excluded suffixes such as looking up a word with pronoun endings. The searching results are also supported with statistics and numbers of occurrences. We have selected the Newspapers category as it covers a wide variety of topics; this category represents a data set containing approximately 135 million words of articles published between 1996 and 2010 in different Arabic countries as shown in Table 3-1.

Paper	**Country**	**Year**
Al-Masri Al-Yawm	Egypt	2010
Al-Ghad	Jordan	2011
Al-Watan	Kuwait	2002
Al-Tajdid	Morocco	2002
Al-Ahram	Egypt	1999
Al-Hayat	London	1996-1997
Al-Thawra	Syria	Unknown

Table 3-1: Sources of the articles in the Newspaper category.

[6] http://arabicorpus.byu.edu/index.php

3.3.1 Composites of the Patterns

We have initially constructed our set of patterns using a series of different kind of tokens separated by spaces. The tokens have been made easy to understand so that the set can be readily modified and extended with new patterns. For the pattern-matching process, a separate algorithm would convert each pattern in the set into sequences of literal characters and special symbols namely regular expressions that obey the conventions used by the JAVA programming language. Tokens used to formulate the patterns comprise the following items:

- ***A Particular Word:*** This type of token search the input sentence for any word that has the same characters as the token under scrutiny. For example, the words "من" and "اجل" in pattern P (1).
- ***Subpattern Reference:*** It is preceded by the (&) sign and refers to a predefined set of (words, phrases, particles). For instance the subpattern ***&This*** in pattern P (1) refers to a list of definite demonstrative nouns (...تلك ، ذلك ،هذه ،هذا).
- ***Part-of-Speech Tag:*** Indicated in patterns by uppercase characters. Each tag represents a certain syntactic category assigned to each word in the input sentence such as the definite noun tag *DTNN* in pattern P (1). Part of Speech (POS) tagging has been obtained from the Stanford tagger system. The POS tagger's developers have reported that it works rapidly with per token accuracies of slightly over (97%). POS labels are listed in Table 3-2.
- ***A Slot:*** This token reflects the adjacent words that represent the *cause* or the *effect* part of the relation under scrutiny; it is indicated by the characters [C] or [E] respectively.
- ***A Symbol:*** Instructs the *Pattern Recognizer* model to take specific action during the pattern matching procedure. These symbols could be one of the following:
    - (+): Instructs the *Pattern Recognizer* to add the matched token followed by such symbol to the *cause* slot. For example, the plus symbol in pattern P (1) implies that the word "اجل" has to be included in the *cause* slot.
    - (++): This symbol has the same action as for the symbol (+) except that the identified token is added to the *effect* slot. For example, the two plus sign in

pattern P (1) intends that any word matches the POS tag - past tense - has to be added to the *effect* slot.

- (@): Any token followed by this symbol instructs the *Pattern Recognizer* to accept all possible suffixes (…ون،ات،هم،ها) that could be bound after. If this symbol is located alone, it indicates that any word ending with pronouns will be accepted.
- (#): Any token followed by this symbol instructs the *Pattern Recognizer* to accept all possible prefixes (ن ،ت،ي) that could be bound before.
- ($): Instructs the *Pattern Recognizer* to match the word under scrutiny against a specific verb template where the (w) character represents the basic units of the Arabic root (ل /ع /ف). For example, the token **$Awww** in pattern P (2) matches any word that has the template (أفعل), the same for the token **$MwAww** which matches the template (مفاعل).
- (/): Separates a number of alternative tokens that the *Pattern Recognizer* has to look for.
- (^): This symbol precludes a certain word from being matched i.e. if the word under scrutiny matches a token followed by this symbol; the pattern matching process indicates a mismatch pattern.
- (W*n*): Instructs the *Pattern Recognizer* to match at most *n* occurrences and at least one occurrence of adjacent words, i.e. the *W3* token will match one or two or three words.
- (C): This is a wildcard symbol indicating the *Pattern Recognizer* to match any number (excluding zero) of adjacent words or phrases.
- (!): Instructs the *Pattern Recognizer* to normalize the word under scrutiny before it is matched against the token. Normalization is discussed in Section 5.2.
- ( ): Locating two braces implies that it is optional to match the token contained within.

P (1) R (&C) [C] AND + من +اجل &This (DTNN) ++VBD [E] &.

P (2) X C $Awww مما C

Tag	Description	Tag	Description
CC	Coordinating conjunction	NNS	Noun, plural
CD	Cardinal number	PRP	Personal pronoun
DTNN	Determined Noun	RB	Adverb
FW	Foreign word	RP	Particle
IN	Preposition conjunction	SYM	Symbol
JJ	Adjective	VB	Verb, base form
MD	Modal	VBD	Verb, past tense
NN	Noun, singular	VBP	Verb, non-3rd person
NNP	Proper noun, singular	WP	Wh-pronoun
NNPS	Proper noun, plural	PUNC	Punctuation

Table 3-2: Part Of Speech tags.

3.3.2 Establishing the linguistic patterns

The *Pattern Recognizer* model generally makes use of the same techniques as have been used by (Khoo et al., 1998) for identifying and extracting *Causal* relations in English language. Since the aforementioned discourse connectives are functional linguistic devices that acquire meaning from context, the constructed patterns should be adapted to cover various phrasing of sentences and syntactical structures. The pattern development process went through several steps of reasoning methods. *Inductive* and *deductive* phases have been assembled into a single circular one so that the patterns continually cycle between both of them until we end up developing a set of approximately 900 general patterns. In the remainder of this chapter, we explain the approach of constructing patterns indicating causation which work similarly for constructing patterns indicating explanation.

- ***Inductive Phase***: It is the initial step of the development process which involves making specific observations from a sample of sentences containing *Causal* relations. This implies detecting regularities and features that indicate the presence of a *Causal* relation. This phase has led us to formulate some tentative patterns specifying *cause* and *effect* slots.

For example pattern P (3) has been constructed from sentence (5) specifying that the words preceding (نظرا) represent the *effect* slot while the words following (نظرا) represent the *cause*.

(5) أجلت *ناسا* أمس هبوط مكوك الفضاء اتلانتيس وذلك نظرا لسوء الأحوال الجوية.

"NASA postponed the landing of the space shuttle Atlantis yesterday ***due to*** bad weather."

P (3) R (&C) [E] AND ذلك +نظرا [C] &.

- ***Deductive Phase:*** Involves exploring the patterns that have been formulated in the previous step by testing them against text fragments extracted from the corpus. Each text fragment has contained an occurrence of the causative unit addressed by the pattern and a "window" of 10 words before and 10 words after this occurrence. The Arabic writer, however, prefers the use of regrouped and large grammatical chunks. Hence, in many cases a longer "window" has needed to be investigated.

Three types of errors may be returned upon conducting the patterns test in the *deductive* phase. Each kind of error has been handled by performing another *inductive* step. Errors found can be classified as follows:

1. ***Undetected Relations***: This error occurs when the constructed patterns are unable to locate the presence of a *Causal* relation in a text fragment. To fix this error, more patterns need to be added so that the missing relation can be identified. In some cases it may be better to modify a pattern to cover all the absent relations by omitting some of its features so that it is shifted up from the more specific pattern to a more general one.

For example, pattern P (3) that has been previously constructed to identify the *Causal* relation in sentence (5) would obviously miss the *Casual* relation presented in sentence (6), because of omitting one feature of pattern P (3) which is the word "نظرا". For that we have created pattern P (4) that is able to retrieve the missed relation.

(6) اولت الحكومة اهتماما كبيرا لتطوير القطاع الزراعي في الاونة الاخيرة و ذلك رغبة منها بتحقيق الامن الغذائي .

"The government has recently paid great attention to the development of agriculture ***to*** achieve food security"

P (4) R (&C) [E] AND ذلك [C] &.

2. ***Irrelevant Relation:*** This is linked to the situation when the constructed patterns improperly recognize a relation as a *Causal* one. For this kind of error, we need to narrow down the scope of these patterns from the more general into the more specific by adding more constrains on them. Another way to amend this fault is to add a new pattern associated with the void value to exclude the expression that causes the error.

For instance, the word "لذلك" in sentence (7) distinctly expresses causality, so pattern P (5) would correctly indicate the presence of a *Causal* relation. However, the occurrence of the word "لذلك" in sentences (8) and (9) acts as cataphoric and anaphoric references that refers to other elements in the two sentences. This function can be identified by the definite noun following the causative connectors for sentence (8) or due to the connector position - at the end of the sentence- in sentence (9). In both instances new patterns P (6) and P (7) of a void value should be constructed in order to indicate irrelevant relations. It is important to note that sentence (8) still contains a *Causal* relation signalled by the causation *faa* as it will be discussed later in Section 3.4.

(7) يمكن للغبار أن يغطي مناطق التهوية مما يؤدي إلى ارتفاع درجة حرارة الحاسوب لذلك يجب أن تحترس من الغبار.

"Dust can obstruct the ventilation areas of a computer leading to a rise of temperature; ***therefore*** you must protect against dust"

(8) إقرأ نشرة الدواء بعناية قبل تناول أي جرعة منه فقد لايكون لذلك الدواء أي علاقة بمرضك.

"Read the drug leaflet carefully before taking it since ***that*** drug may not be adequate to your illness"

(9) لم يكشف قائد الفريق عن رغبته الاستغناء عن بعض اعضاء الفريق ولكن تصرفاته تشير لذلك.

"The team leader has not disclose his intention to dismiss some of the team members, but his behaviour points out to ***that***"

P (5) R (&C) [C] لذلك [E] &.

P (6) X C لذلك DTNN C

P (7) X C لذلك &.

3. ***Misidentify Slots:*** In some cases, even though a relevant relation is correctly extracted, the pattern fails to fill the cause-effect slots properly. A good remedy for this defect is to reorder the patterns in a way that more specific patterns have the priority over the more general ones.

For example, pattern P (5) is unable to correctly fill the *cause* and the *effect* slots of the *Causal* relation in sentence (10). Therefore, an additional pattern such as pattern P (8) is needed to be created and inserted before pattern P (5).

(10) يعاني الميزان التجاري للسلع من الخلل و لذلك فإن الحكومة بدأت باقامة المشروعات التي تعتمد على الخدمات

"The Goods Trade Balance undergoes some flaws; therefore, projects that rely on public services have been established by the government"

P (8) R (&C) [C] (AND) فإن لذلك [E] &.

Examples of the linguistic patterns for identifying the *Causal* relations signalled by the word "نتيجة" are given in Table 3-3.

Status	Pattern
X	C &Not (W) نتيجة C &. e.g. إن ما وقع بين المدرب واللاعبين لم يكن نتيجة انفعال لحظي.
X	C &Whatever نتيجة C &. e.g. .... واشاروا الى انهم سيخوضون المسابقة ايا كانت نتيجة القرعة التي ستسحب الاسبوع القادم.
X	C $Awww نتيجة L C &. e.g. انتهى الشوط الاول بفارق 17 نقطة وهي افضل نتيجة للمنتخب في نهائيات هذه البطولة.
R	(C) (AND) نتيجة (من) لكن/كانت/كان (&This) [C] ان! [E] &. e.g. ... ، وكان من نتيجة النمو الهزيل للصادرات ان اصبح ترتيب الدولة متاخرا بين الدول المصدرة.
R	(C) (AND) نتيجة (من) لكن/كانت/كان (&This) [C] Verb++ [E] &. e.g. .....، لكن نتيجة تلك الشكاوي المتكررة أنهيت عقود العمال المؤقتة وفصلوا من الشركة.
X	C IN نتيجة C &. e.g. هناك وسائل اخرى إذا كنا نريد أن نؤثر في نتيجة هذه الاختبارات.
R	(&C) [C] (AND) Verb (W3) @ان! نتيجة لذلك/لهذا Verb++ [E] &. e.g. ذكر تقرير صادر أمس أن سياسة الاغلاق طبقت بشكل متزايد في السنوات الأخيرة واشار التقرير أنه نتيجة لذلك انخفض مستوى معيشة المواطن.
R	(&C) [C] (AND) Verb (W3) @ان! نتيجة لذلك/لهذا &Res [E] &. e.g. قالت اللجنة أن اليابان اخفقت في توفير حماية كافية لحقوق الملكية الفكرية للتسجيلات الاوروبية التي تباع في السوق اليابانية وذكرت اللجنة أنه نتيجة لذلك فقد اصبح كثير من اشهر الاصدارات الاوربية لا تتمتع بحماية في اليابان.
R	(C) AND نتيجة [C] VBD++ [E] &. e.g. ...، ونتيجة لارتفاع الكثافة السكانية في هايتي وانهيار البنية التحتية فيها اصبحت عرضة لتأثيرات الكوارث الطبيعية كالفيضانات والأعاصير.
R	(&C) [C] Verb @ان!/ان! نتيجة Verb [E] &. e.g. ....فمرض "جيلان بارى" يمكن أن يحدث نتيجة الإصابة بالانفلونزا أو التسمم الغذائي.

Table 3-3: Some of the patterns involving the word "نتيجة".

ALGORITHM 3-1: Converting a linguistic pattern into a regular expressions string

```
Input: A linguistic pattern.
Output: The equivalent regular expression string.
1.  Replace [c] and [E] symbols with “(\b\w+/\w+\b)+”;
2.  Replace all pair of braces with “()?” ;
3.  Replace all POS Tags (tag) with “(\b\w+/tag\b)”;
4.  Replace all (/) symbols with “|”;
5.  Replace all C characters with “(\b\w+/\w+\b)+”;
6.  Replace all (Wn) symbol with “(\b\w+/\w+\b){1,n}”;
7.  If a token starts with (#) symbol
8.     Add the string “(ت|ي|ن)?” to the beginning of the token;
9.  If a token ends with (@) symbol
10.    Add the string “(ه|ات|هم|ها)?” to the end of the token;
11. If a token starts with (&) symbol
       Retrieve the list of the words and phrases referred
       to by the token and replace it with the token as a
       one set of alternative strings;
12. If a token starts with $ symbol
13.      Replace all w characters with “\w”;
14.      Replace all A characters with “أ”;
19.      Replace all a characters with “ا”;
20.      Replace all Y characters with “ي”;
21.      Replace all W characters with “و”;
22.      Replace all M characters with “م”;
23.      Replace all Q characters with “ة”;
24.      Replace all y characters with “ى”;
25.      Replace all N characters with “ن”;
26.      Replace all C characters with “ء”;
27.      Replace all E characters with “إ”;
28.  End If
29.  If a token starts with (!) symbol
30.      Replace all (آ,أ,إ) with “ا” ;
31.      Replace all (ى) with “ي” ;
32.      Replace all (ة) with “ه” ;
33.  End If
34.  Replace all white spaces with “\s”;
35.  Omit all previous symbols from the string;
36.  Convert all Arabic letters into the equivalent
     UTF-16 encoding characters;
37.  END
```

ALGORITHM 3-1 describes the actions taken to convert the patterns formulated in this study into their equivalent regular expressions. The symbols, characters and operators adopted for generating regular expressions strings are presented in Appendix II.

The algorithm replaces each of the pattern tokens with the appropriate string in order to match the POS tagger output. The tagger produces a sequence of tagged words each of which has the form *word/tag*. For example, line 3 locates all POS tags in a pattern and substitutes each with a string begins with boundary character "\b" followed by a word character "\w" attached to "+" operator in order to match one or more occurrences of any Arabic letter; then the targeted POS tag *"/tag"* is bound to the string followed by another word boundary "\b".

In lines [12-28] the algorithm replaces the symbols that represent Arabic word templates with actual Arabic letters. Finally, the algorithm omits all special symbols and maps all Arabic characters in a pattern with the equivalent encoding character UTF-16. The UTF-16 encoding for the Arabic letters is given in Appendix III. Applying ALGORITHM 3-1 to pattern P (9), generates the converted pattern P (10).

P (9) R (C) AND نتيجة [C] VBD++ [E] &.

P (10) R (\b\w+/\w+\b\s)*\u0648\s?\u0646\u062A\u064A\u062C\u0629\s(\b\w+/\w+\b\s)+
\b\w+/VBD\s(\b\w+/\w+\b\s)+(\b\W/PUNC|CD|SYM\b)

3.4 Justification Particles

The justification particles are those types of letters that are prefixed to certain word to indicate causation and explanation in sentences; this set of particles includes ***purpose lam*** (لام التعليل), ***causation faa*** (فاء السببية) and ***causation baa*** (باء السببية). However, these particles are highly ambiguous since they hold a wide range of functions and purposes other than causation or explanation. Therefore, linguistic patterns cannot be employed for the detection of the syntactical rules that govern them. Alternatively, each of which requires specific actions and procedures to be taken into consideration.

The issue here is that to precisely recognize the justification role of these particles requires an accurate syntactic parser which has not been used in this study. Hence, we have proposed three algorithms that aim to make a judgment on whether a word starting with any of these

particles implies a justification function. These algorithms do not always precisely identify the justification role of the aforementioned particles, but they effectively work with very little computational expense.

3.4.1 Purpose Lam (لام التعليل)

Purpose Lam is one of the most complicated particles in the Arabic language as it expresses many meanings insomuch that some grammarians count more than 30 different purposes of it. For instance, ***lam of denial*** (لام الجحود) as in لم يكن خالد ليشرب الحليب "Khalid was not a man to drink milk" and ***lam of possession*** (لام الملك) when it indicates the right of property as in كان لأحمد سيارة كبيرة "Ahmad had a large car".

However, our concern here is the case of *lam at-'taleel* which is originally a preposition implies the intention of the agent. *Lam at-'taleel* may also indicate the purpose for which, or the reason why, a thing is done. In this context, the Arab grammarians take *lam-at-'taleel* to function similarly to (لأن) or (لكي) (Wright and Caspari, 1896).

The procedure we propose to recognize *lam at-'taleel* is outlined in ALGORITHM 3-2. It accepts as input a word (*W*) prefixed with the particle "*lam"* along with the tagged sentence that the word belongs to and a list of stop words. As output it returns a true value if the word's context suggests a justification role and false otherwise.

In the first line the algorithm checks if the word's length including the "*lam"* character is less than four letters, in which case the word is a particle such as "لم ، لقد ، لن،...". It also checks if the word is contained in the stop words list; if yes it yields a false result.

In lines [5-8] the algorithm inspects the POS tag assigned to the word, if the syntactic category of (*W*) is in the set (proper noun, singular noun, plural noun and preposition) the algorithm returns false. Then the algorithm treats the case of double "*lam"**;*** it examines that the syntactic category of the word following (*W)* is a preposition, if not a false value will be returned. The double "*lam*" in sentence (11) is an example of a false case.

In line 13 the algorithm returns true if (*W)* matches any form of the verbs category. The next step tests if (*W)* has the template (أفعل), at this point we exclude the cases when "*lam"* prefixes

اسم التفضيل *"noun of preeminence"* as in sentence (12). The condition in line 17 eliminates the words that denote plural nouns to both genders.

In line 19 the (*W)* is reduced to its stem before it is checked against a set of nominal templates, those templates refer to اسم الفاعل *"present participle"* and some forms of جمع التكسير *"broken plural/irregular plural"*, if (*W)* belongs to any of the former templates the algorithm returns false.

In lines [21-24] the algorithm considers the case when (*W*) length is more than four characters and starting with (م) letter; it searches the (*W)* for the occurrence of (ا) letter and returns true if it is located, otherwise it returns false. This way we exclude the following forms of nouns: اسم المكان *"noun of place"* such as the one in sentence (13), اسم الزمان *"noun of time"* and اسم الالة *"noun of instrument"* as in sentence (14). However, if a word of the previous forms contains (ا) letter, it becomes in the infinitive form expressing justification as in sentence (15).

Finally, in case that the aforementioned *if* statements were not applicable the algorithm returns a true value recognizing (*W)* as a justification indicator.

(11) بالنسبة للإسكان فقد أكدت الحكومة أنها بصدد اعداد دراسة شاملة.

"As for the housing issue, the government has confirmed that it is considering a comprehensive study in this regard."

(12) وصلت البعثة إلى المطار بعد انتظار دام لأكثر من عشر ساعات.

"The delegation eventually arrived at the airport after waiting for more than ten hours."

(13) سمحت وزارة الصحة لمعمل الالبان بإستئناف نشاطه.

"The Ministry of Health allowed the dairy factory to resume its operations."

(14) ذكر الأديب أن العديد من أعماله تعرضت لمقص الرقيب.

"The author mentioned that many of his works were subject to censorship."

(15) يجتمع وزراء البيئة الشهر المقبل لمناقشة تقليل انبعاث الملوثات.

"Ministers of the Environment will hold a meeting next month to discuss ways of reducing the emissions of pollutants."

ALGORITHM 3-2: Determining the potential justification function of the particle "*lam*".

```
Input: A Word W starting with the character Lam.
       The tagged sentence in which W appears.
       Stop words list.
Output: Determination of whether W constitutes a justification
        relation?
1.  If (W length< 4) OR (W contains in Stop words list)
2.     Return false;
3.  If the word preceding W is VBP or Proposition
4.     Return false;
5.  If W tag is a Proper Noun
6.     Return false;
7.  If W tag excluding lam character is (NNP or NNS or IN)
8.     Return false;
9.  If the second character of W is lam
10.       If the word after W is a proposition
11.          Return true;
12.         Else return false;
13. If W type is a verb
14.    Return true;
15. If W has the template (أفعل)
16.    Return false;
17. If last characters of W are in the set{يه ،ية ، ات،ين، ون،يا}
18.    Return false;
19. If stemmed W matches any of the following templates
         {فعيل،مفاعل،فاعل،مفاعيل، أفعال،مفعال}
20.     Return false;
21. If (W starts with (م) character) && (W's length>4)
22.      If (ا) character is found
23.         Return true;
24.      Else return false;
25. Return true;
26. END
```

3.4.2 Causation Faa (فاء السببية)

The particle "*faa*" is also considered a challenging particle since it plays a multifunctional status and has many semantic properties. The illustrative examples stated in this discussion were taken from (Saeed and Fareh, 2006). One of the particle "*faa*" roles is to signal a consequential relationship between two elements or events occurring consecutively and in the

order indicated in the sentence. For example, قام خالد فاحمد "Khalid stood up then Ahmad". Also, "*faa"* has an adversative function in which it expresses a contrast between two clauses, the second of which stands in adversative relation with the preceding. The following example illustrates this function دعاني صديقي لزيارته فلم اجب دعوته "my friend invited me to visit him, but I turned down his invitation". In addition, "*faa"* has a significant role that is directly related to the purpose of this study in which it contributes to indicating causation between two parts of sentence. Consider the two examples in sentences (16) and (17).

(16) احب احمد المسرح فابدع فيه.

"Ahmad loved theatre and ***so*** he excelled in it."

(17) لا تبك فإن البكاء ضعف.

"Do not cry ***because*** crying is a weakness."

Several newspaper articles from the ***arabiCorpus*** were surveyed in order to identify grammatical and syntactical characteristics that help recognizing the cases in which the particle "*faa*" functions as a causative/resultative conjunction. Consequently, we came up with the set of rules formulated in ALGORITHM 3-3.

ALGORITHM 3-3: Determining the potential causation function of the particle "*faa"*.

```
Input: A Word W starting with the character faa.
       The tagged sentence TS in which W appears.
       Stop words list.
Output: Determination of whether W constitutes a causation
          relation?
1.  If W contains in Stop words list or W's stem starts with
    faa
2.     Return false;
3.  If the word preceding W is VBP or Proper Noun
4.     Return false;
5.  If W tag is a Proper Noun
6.     Return false;
7. If the words (بالنسبة/اما) appear in TS before the occurrence of
   faa
8.   Return false;
9. If W tag excluding faa character is a Proper Noun
10   Return true;
```

```
11. If the word following W starts with (ال)
12. Return true;
13. If W type is a verb
14.    Return true;
15. If W belongs to the set of words {جميع، بعض،كل، قليل...}
16.    Return true;
17. If W is a demonstrative pronoun or a relative noun
18.    Return true;
19. Return false;
20. END
```

3.4.3 Causation Baa (باء السببية)

Another particle that poses many difficulties is the particle "*baa*". Grammarians denote various uses of "*baa*" (Wright and Caspari, 1896). One use of this particle is "الظرفية" to express time and place, for example, "سافر قبلي بيومين" "*He travelled two days before me*". Another use for "*baa*" is to indicate adhesion "الإلصاق" as in "لان الدود يتعلق بالثمار" "because worms stick to the fruit". It can also be used to form negation expressions as in "لست بعالم" "I don't know". Moreover, it expresses the reason, cause or explanation such as the particle "*baa*" in two sentences (18) and (19). ALGORITHM 3-4 attempts to recognize this role of the particle "*baa*".

(18) يرزقه الله الصبر ببركة دعائه.
"God will grant him patience through the salutary power of prayer to him"

(19) كتبت بالقلم
"I wrote with the pen"

ALGORITHM 3-4: Determining the potential causation function of the particle "*baa*".

```
Input: A Word W starting with the character baa.
       The tagged sentence TS in which W appears.
       Stop words list.
Output: Determination of whether W constitutes a causation
        relation?
1.  If W contains in Stop words list or W's stem starts with
    baa
2.     Return false;
3.  If W's tag is (Proper Noun or plural noun)
4.     Return false;
```

```
5.  If W precedes with a negative particle
6.     Return false;
7.  If W excluding baa is indefinite noun and the word
    preceding W is not a verb
8.     Return true;
9.  If the word preceding W is a definite noun
10.    Return false;
11. If the word following W is a Verb or Preposition or
    pronoun
12.    Return false;
13. If last characters of W or the word following W belong to
    the set {ه ، هم ، همـا ، هن ، هـا ، نـا}
14.      Return true;
15. If W excluding baa or the word following W starts with (ال)
16.    Return true;
17. Return false;
18. END
```

3.5 Combining Relations

It is a common trait of natural languages that a text involves a sequence of events that leads up to some final effect; this causal/explanatory chain results in combining relations. Let us consider the three events subsumed in text (20), we notice that event 1 in slot I causes event 2 in slot II to form the *Causal* relation C_1-E_1. Similarly, event 2 causes event 3 in slot III creating the *Causal* relation C_2-E_2. However, event 1 is also responsible for the result occurring in event 3. Accordingly, a new *Causal* relation i.e. C_3-E_3 is created where event 1 and event 2 are joined together constituting the *cause* part of the new relation, and event 3 constitutes the *effect* part. The formula (3-1) illustrates this rule of relations combination.

(20) $^{C3}[$ $[$ $[^{I}$إن موقع البلاد العربية القريب من خط الاستواء$^{C1}]$ يجعل$[^{II}$أشعة الشمس تدخل الطبقة الجوية العليا بشكل عمودي$^{E1}]^{C2}]$ مما يجعل $[$ $[^{III}$أشعتها المسرطنة والمخربة للبشرة أكثر نفاذا وخطرا بخسمة أضعاف من تلك التي تستطع في أوروبا وشمال أمريكا.$]^{E2}]^{E3}$

"In the Arab countries which are close to the equator, sun rays vertically permeate the upper atmosphere, and this makes the sun's carcinogen and skin-damaging rays five times more permeable and dangerous than the sun that shines in Europe and North America."

If: [C1 – E1] & [C2 – E2] where E1 = C2 $\Rightarrow$ [C1,C2 - E2] (3-1)

3.6 Summary

This chapter described our work in identifying semantic relations occurring within Arabic sentences. More specifically, the two intrasentential relations of *Causal* and *Explanatory* have been under consideration. In doing so, a set of linguistic patterns have been constructed based on syntactic and morphological features. The patterns are employed by the *Pattern Recognizer* model so that it extracts cause-effect and method-effect information. This information is very important for QA system targeting "*why*" and "*how to*" questions.

In addition, three algorithms have been introduced to boost the effectiveness of the *Pattern Recognizer* by discovering the causal/explanatory role of the justification particles which was another concern of this chapter.

The next chapter addresses the task of relations extraction at the sentence level. It proposes a new methodology that attempts to deal with the problem of computational complexity associated with the text derivation process.

Chapter 4

Automatic Text Structure Derivation

4.1 Introduction

The processing of complex questions with explanatory answers such as "*why*" and "*how to*" involves searching texts for arguments mainly *Causal* and *Explanation* relations. The previous chapter was dedicated to describing the method adopted to build the *Pattern Recognizer* model where a set of linguistic patterns were constructed. In doing so, the model is able to identify the presence of causality and explanation in a single sentence (intrasentential relations). This set, in turn, makes a fundamental contribution to recognize potential answers in systems addressing "*why*" and "*how to*" questions.

The main issue arising at this point is that arguments might be distributed over several sentences, making it necessary to acquire a proper linguistic knowledge about the presence of relevant relations in text. Therefore, a discourse analysis approach able to automatically derive text structure needs to be incorporated to discover *Causal* and *Explanation* relations among sentences (intersentential relations).

The structure of texts can be visualized as multiple sentences which are related to each other. Such combination is called a discourse which in itself consists of multiple discourse segments, non-overlapping spans of text, or a complete sentence. The coherence between these segments is provided by rhetorical relations. A discourse segment can for example provide additional information about a preceding segment.

Much attention has been given to developing technologies capable of building up a rhetorical structure and presenting explanations based on the text structure. It is considered useful for many natural language applications that include speech and image generation (Lindley et al., 2001), text summarization (Marcu, 2000b), essay scoring (Burstein and Marcu, 2003) and machine translation (Ghorbel et al., 2001). There are many theories that have been introduced to identify coherent relations in texts as the one proposed by Grosz and Sidner (1986), Rhetorical Structure Theory (RST) (Mann and Thompson, 1988), the Graph Bank Model (Wolf and Gibson, 2005), and the Segmented Discourse Representation Theory (SDRT) (Asher and Lascarides, 2003).

RST is a well-established approach for discourse analysis and studies have shown it to be a very effective tool in many computational linguistics applications (Taboada and Mann, 2006). Moreover, human annotators show considerable agreement when using it, which indicates that the authors of the theory have clearly defined the rules and the guidelines for segmenting and selecting rhetorical relations. This chapter starts with a general introduction of the RST. A description of the methodology proposed to derive discourse structure follows, and finally the chapter ends with providing a worked example.

4.2 Review of Rhetorical Structure Theory

4.2.1 Overview of RST

RST has been first developed by Mann and Thompson in the 1980s as a result of exhaustive analyses of English texts. RST is primarily aimed at describing those functions and structures that make text an effective and comprehensible tool for human communication (Mann et al., 1993).

Based on their observation of edited texts from a wide variety of sources, Mann and Thompson (1988) have made several assumptions about how written text functions, and how it involves and uses words, phrases and grammatical structure as summarized below (Mann et al., 1992):

● **Organization:** Text consists of functionality significant parts; the parts are elements of patterns in which they are combined to create larger parts and whole texts.

● **Unity and coherence:** There must be sense of unity to which every part contributes.

● **Hierarchy:** Elementary parts of a text are composed into larger parts, which in turn are composed of yet larger parts up to the scale of the text as a whole.

● **Relation Composition:** Relations hold between parts of a text. In which every part of a text has a role, a function to play, with respect to other parts in the text. A small finite set of highly recurrent relations holding between pairs of parts of text is used to link parts together to form larger parts. All rhetorical relations that can possibly occur in a text can be categorized into a finite set of relation types.

● **Asymmetry of Relations:** RST establishes two different types of units. Nuclei are the most important parts of a text, whereas satellites contribute to the nuclei and are secondary.

RST addresses text organization by means of relations that hold between units of text (spans) called rhetorical relations. Spans range in length from clausal or sub-clausal units to the text as a whole. Every span of a text has a role, *nucleus* or *satellite,* with respect to other spans in the text. Nuclei are the most important parts of a text whereas satellites contribute to the nuclei and are secondary.

All rhetorical relations that can possibly occur in a text can be categorized into a finite set of relation types. The most common type of text structuring relation is an asymmetric class, called nucleus-satellite relations, in which the nucleus is considered to be the basic information, and more essential to the writer's purpose than the satellite. The satellite contains additional information about the nucleus and it is often incomprehensible without the nucleus, whereas a text where the satellites have been deleted can be understood to a certain extent.

Based on their observation, Mann and Thompson have defined 24 rhetorical relations considered classical RST relations, and six more relations have been added to produce a total of 30 extended RST relations (Mann and Taboada, 2005). Table 4-1: illustrates some of the relations identified by Mann and Thompson.

Relation definition consists of four fields specifying particular judgments that the text analysts or writers have to make in building RST structure (Mann and Taboada, 2005). Table 4-2 shows the definition of the *Condition* relation as it appears in (Mann et al., 1993).

Relation Name	Nucleus	Satellite
Background	Text whose understanding is being facilitated	text for facilitating understanding
Elaboration	basic information	Additional information
Antithesis	ideas favoured by the author	ideas disfavoured by the author
Enablement	An action	information intended to aid the reader in performing an action

Table 4-1: A sample of the relations used in RST.

Definitional Element	Observer's Finding
Constraints on the nucleus, N:	None.
Constraints on the satellite, S:	S presents a hypothetical, future, or otherwise unrealized situation (relative to the situational context of S).
Constraints on the N + S combination:	Realization of the situation presented in N depends on realization of that presented in S.
The effect:	R recognizes how the realization of the situation presented in N depends on the realization of the situation presented in S.

Table 4-2: Definition of the *Condition* relation.

Schemes are being used to visualize the text structure in RST. Each schema indicates a specific kind of text structure and how it is decomposed into other text spans (Mann and Thompson, 1988). In every schema, there are horizontal lines representing text span and vertical or diagonal lines representing identifications of the nuclear spans. The arrows link the satellite to the nucleus of a rhetorical relation. The relations are represented by curved lines labelled with the name of the rhetorical relation that holds between the two units over which the relation spans. Figure 4-1 presents two schemas taken from (Mann and Taboada, 2006) as

examples of the *Concession* and *Contrast* relations. The *Concession* relation scheme represents the nucleus – satellite relation type where the nucleus "we shouldn't embrace every popular issue that comes along" is considered to be the core information and more central than the satellite "Tempting as it may be,". On the other hand, the *Contrast* relation is of a multinuclear relation type joins two units that seem to be of equal importance. There are basically five types of schemas where arcs point at nuclei, whereas straight lines indicate text spans in multi-nuclear relations as shown in Figure 4-2 (Mann and Thompson, 1988).

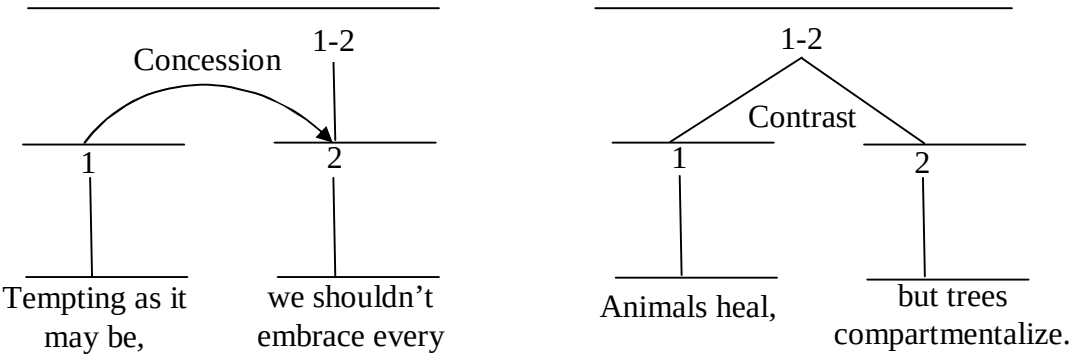

Figure 4-1: Concession and Contrast relations (Mann and Taboada, 2006).

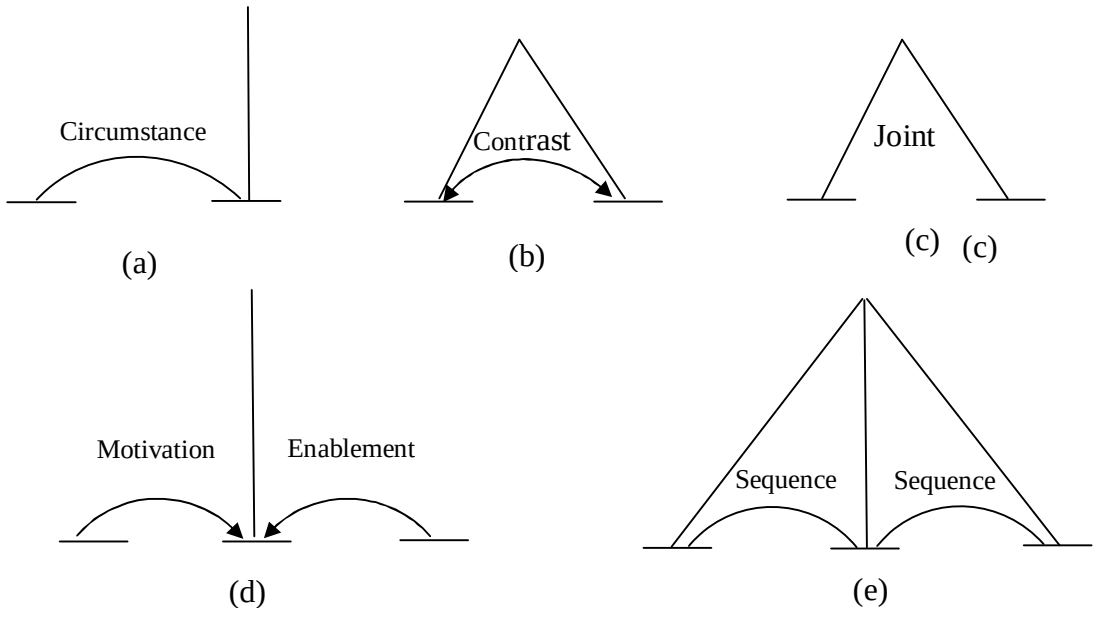

Figure 4-2: The basic types of RST schemas (Mann and Thompson, 1988).

The smallest text spans that hold rhetorical relations are named Elementary Discourse Units (EDUs). Two or more EDUs together can form a new span, which again holds a rhetorical relation with another text span. This way, a hierarchical structure is created for each text. Figure 4-3 presents an example of discourse structure resulted from applying RST to a Scientific American article (Mann and Taboada, 2005).

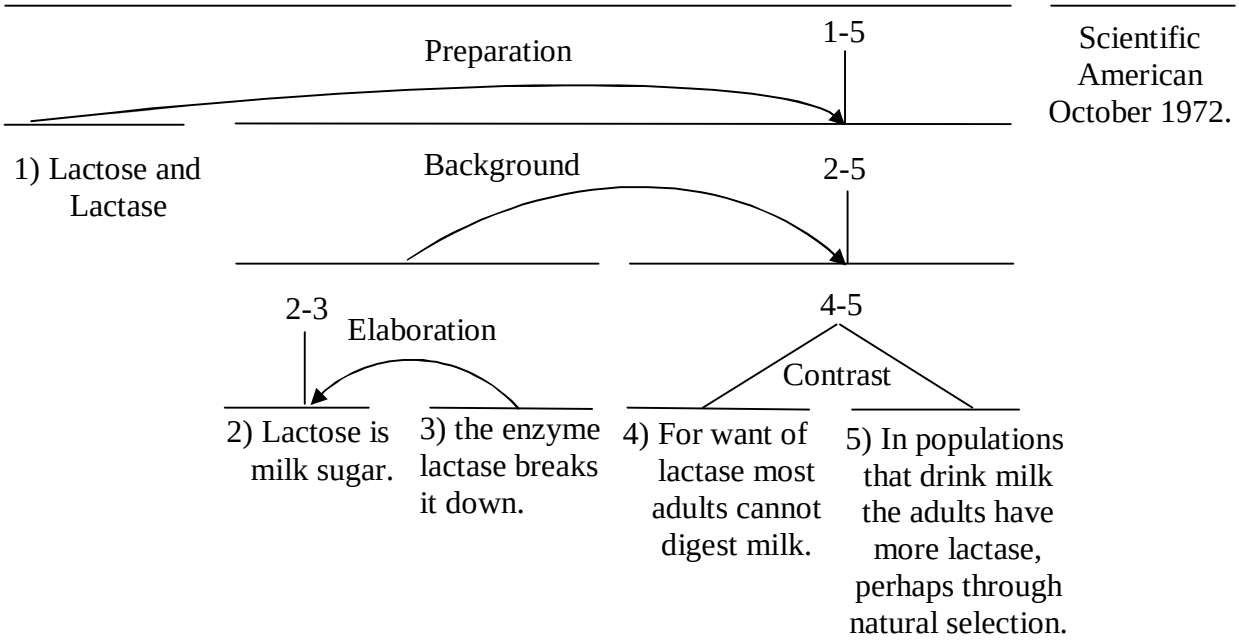

Figure 4-3: An example of the outcome of RST (Mann and Taboada, 2005).

Increasingly, RST is being used as a tool for analyzing the structure of natural language texts. Furthermore, RST has proven to be adequate in computational implementations, in the automatic analysis of texts and in the generation of coherent text (Mann and Taboada, 2006).

4.2.2 Employing RST for Arabic Question Answering

Since answers to "*why*" and "*how to*" questions are argumentative fragments of text that are expected to be rhetorically related to what is questioned, it is essential to exploit rhetorical relations in order to recognize potential answers in texts. The distinction that RST makes between the part of a text that realizes the primary goal of the writer, termed nucleus, and the part that provides supplementary material, termed satellite, makes it an appropriate tool for analyzing argumentative paragraphs.

Consider the following example which explains the method used to extract answers. Text (21) is broken into seven elementary units delimited by square brackets, a rhetorical analysis of the text is shown in Figure 4-4.

(21) *[حذر بحث علمي حديث من أن قناديل بحر عملاقة قد تهيمن على محيطات العالم][1] [جراء الصيد الجائر والتغيرات المناخية وأنشطة بشرية أخرى قد تؤدي لفناء الثروة السمكية.][2] [وتحذر دراسة أجراها "مركز CSIRO للأبحاث البحرية والجوية" الأسترالي، من نوع ضخم من قناديل البحر، يدعى "نورمورا Normura" وله قابلية النمو ليصل حجمه إلى حجم مصارع سومو ياباني، وقد يزن 200 كيلوغرام، بقطر يبلغ المترين.][3] [ويعمل باحثون على تجربة تقنيات مختلفة للسيطرة على انتشار قناديل البحر،][4] [منها استخدام الموجات الصوتية لتفجير تلك المخلوقات التي تتميز بجسم شفاف ، وتطوير شبكات خاصة للقضاء عليها.][5] [ويعزو الباحثون التزايد الهائل في أعداد قناديل البحر][6] [للصيد الجائر للأسماك التي تقتات على قناديل البحر وتتنافس معها على موارد الغذاء.][7]*

[A new research warns that giant jellyfish may dominate world's oceans][1] [due to overfishing, climate change and other human activities, which could lead to destroy fisheries.][2] [A study led by "CSIRO marine and atmospheric research" in Australia warns of giant jellyfish called "Normura" that can grow as big as a sumo wrestler, they weigh up to 200 kilograms and can reach 2 meters in diameter.][3] [Researchers are experimenting with different methods to control jellyfish,][4] [some of these methods involve the use of sound waves to explode these creatures that have transparent body and develop special nets to cut them up.][5] [Scientists said that the cause of this explosion number of jellyfish][6] [is the overfishing that feed on small jellyfish and compete with them for their food.][7]

Given the following question - of "*why*" type - related to the above text, we need to extract an answer according to the derived schema.

{ ما سبب تزايد اعداد قناديل البحر ؟ }

{What causes jellyfish blooms?}

We notice that the words of the question match unit 6. Also, unit **7** provides the cause of the problem stated in unit 6. This means that an interpretation relation holds between unit **7** and unit 6 which is labelled as *Rel3* in the schema of Figure 4-4. Because of the relevance between the question and unit 6, we can select the correspondent part of the relation, i.e. unit **7**, as a candidate answer**.**

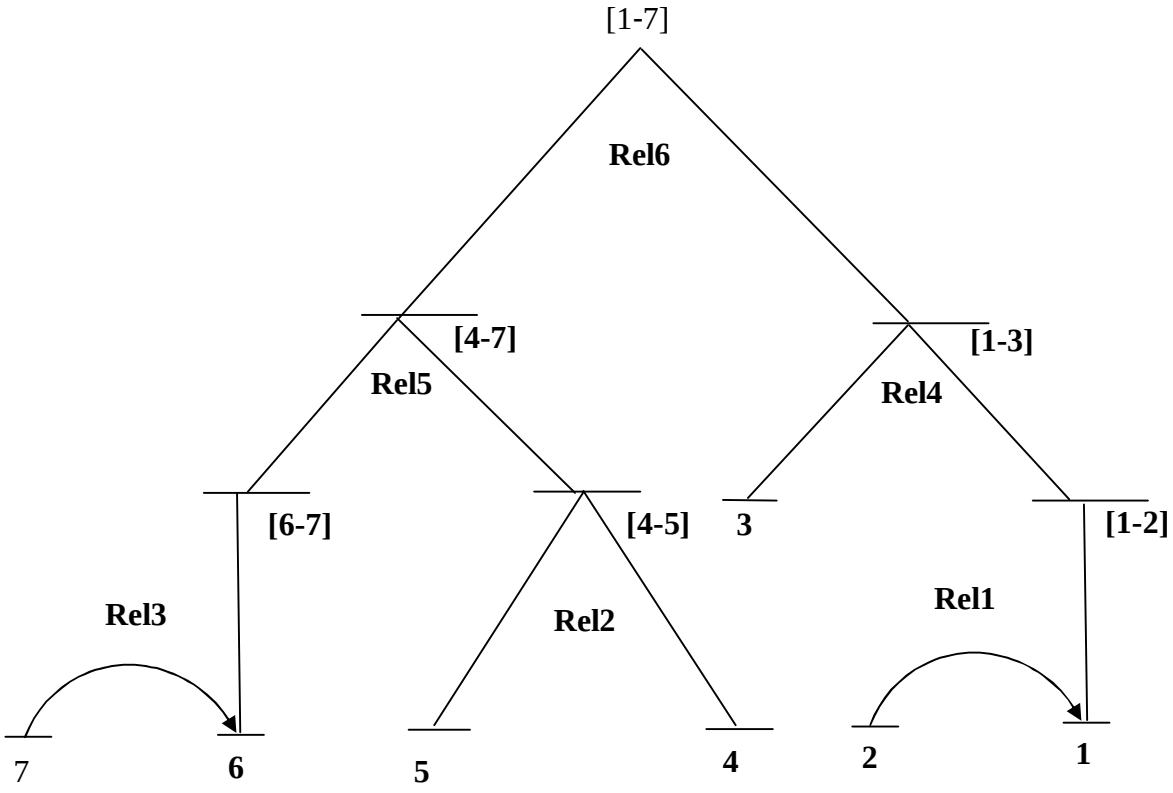

Figure 4-4: A scheme representation of the text.

Now in the case of the following question, belonging to the *how-to* question type

{ كيف يمكن لنا الحد من انتشار قناديل البحر ؟ }

{How do we control jellyfish blooms?}

One can observe that unit 5 gives some methods for solving the problem mentioned in unit 4, so it is concluded that an *Explanation* relation holds between the two units i.e. *Rel 2***.** Since the question corresponds to unit 4, we can select the other part of the relation i.e. unit 5 as a candidate answer.

Thus, rhetorical relations would be a good complementary solution to the pattern-based relations extraction approach presented in Chapter 3.

4.3 Automatic RST Annotation Systems

Writing has always been considered as a complex and demanding activity undertaken by human beings. This is because of the huge variety of linguistic forms the writer may include to achieve his communicative objectives in addition to the tricky nature of the text itself which frequently develops into debatable issues when it comes to grasping these intentions. Accordingly, being able to automatically derive hierarchical structures of this kind of a rich medium is a time-intensive effort.

The literature shows that a number of studies tackling the problem of automatic discourse parsing have been performed in recent years. A fair number of the developed parsers have been eventually applied to summarize texts. The principle behind summarization is that the nuclei sentences are more likely to be retained than the satellites ones; the nuclei are then joined to produce a shorter version of a text. However, the recognition of discourse structure is still a difficult task. In what follows, we will present a general review of the previous automatic RST systems that proposed full structure parsers.

- ***(Simon Corston-Oliver 1998):*** Corston-Oliver (1998) has presented his parser Rhetorical Structure Theory Analyzer (RASTA) to generate n-ary branching trees for unrestricted texts. RASTA exploits resources available within Microsoft English Grammar MEG system in order to get syntactic analyses and logical forms of an input text. Given these forms, the parsed text is then processed through three computational procedures. Firstly, the segmentation process in which the text is divided into EDUs. Secondly, the discovering of all potential relations between each pair of EDUs. This process is carried out in accordance with a number of criteria that have been formulated for each type of relation. Finally, the tree-building process that produces discourse trees based on the relations set that has been hypothesized in the previous step. Oliver has employed a set of 13 rhetorical relations arguing that the restriction to this number of relations is due to computational efficiency considerations; where the smaller the set of hypothesized relations the faster the algorithm for constructing RST trees to test all possibilities. RASTA is an extension of a previous work introduced by Marcu (1996) which has suffered from combinatorial explosion issue - as the number of hypothesized relations increases, the number of possible RST trees increases

exponentially. This is because of the fact that Marcu's algorithm first produces all possible combinations of trees and later rejects a great number of them as ill-formed ones. RASTA has resolved this problem by avoiding tracks that would lead to ill-formed trees in advance so that no need to validate the constructed trees afterwards. To meet this strategy, Oliver has associated each hypothesized relation with weights (heuristic scores) based on linguistic intuition. Thus, RASTA starts considering the relations ranked highest in the possible relations list, it then moves to the second relation in the list and so on. The strategy is to start building up more plausible representations of discourse structure before less plausible ones.

- ***Daniel Marcu (2000):*** Daniel Marcu has proposed a shallow analyzer to employ the formalization of rhetorical relations in RST. He has described it as 'shallow' because it does not use any traditional parsing or tagging techniques. He has used a surface-based approach to decompose a free unrestricted text into EDUs, hypothesizes rhetorical relations that hold among textual units based on the appearance of cue phrases and then, produces all binary rhetorical structure trees compatible with the hypothesized relations (Marcu, 2000a). Assuming that the rhetorical structure of text correlates with the orthographic layout of the text, Marcu has pointed out that the knowledge of discourse markers usage is sufficient to determine the elementary textual units and detection the relations that have discourse function. Whilst in case where no discourse marker could be found, he has exploited text cohesion by using word co-occurrence to measure similarity between two sentences. If this similarity is above a certain threshold, a decision is made to add an *Elaboration* relation between the sentence that comes later and the one that went before or a *Background* relation to relate the sentence that comes before with the next one. Otherwise, a *Joint* relation is assumed to relate the two textual units. A corpus analysis has been performed based on 450 discourse markers and an average of 17 text fragments each. This analysis has led him to extract discourse related information for each cue phrase under scrutiny, e.g. the ***position*** of the discourse marker in the textual unit, the ***rhetorical relations*** that are signalled by the discourse marker, ***where to link*** in order to specify whether the textual unit that contains the cue phrase is related to a unit found before or after it, and ***break action*** that describes where to create an elementary unit boundary in the

input text. Marcu has devised 12 axioms to be used within his algorithm in order to build all valid text structures. These axioms explain how text spans can be assembled into larger spans. The proposed methodology has been evaluated on five American scientific texts; the automatically built trees have been compared with the ones generated manually by two annotators. The overall *recall* for identification rhetorical relations is 40% lower than the *recall* obtained by the human analysts. This is because the text analyzer misidentified a lot of elementary units, whereas the precision obtained for the same task is close to the analysts by 78%.

- ***Radu Soricut and Daniel Marcu (2003):*** Soricut and Marcu (2003) have developed their automatic sentence-level parsing of discourse (SPADE) system based on a Treebank annotated with discourse structures known as RST Discourse Treebank (RST-DT) (Carlson et al., 2002). RST-DT consists of 385 Wall Street Journal articles extracted from the Penn Treebank in which the sentences are associated with syntactic trees. These articles have been manually annotated with discourse structures in accordance to RST formalization. RST-DT has motivated a number of researchers to exploit this annotated corpus as training and evaluation data for the English language. SPADE uses two probabilistic models in order to accomplish the task of sentence segmentation into non-overlapping discourse units and then linking these units with the correspondence hierarchical structures. However, their discourse parser has been restricted to build sub-trees spanning only over individual sentences. With respect to the discourse boundary insertion phase, the statistical model relies on lexical and syntactic features in order to assign a probability value for each word in the input sentence; all words with a probability higher than 0.5 is considered as a boundary marker. Likewise, another probabilistic model has been established to allocate a set of probabilities to each potential discourse tree among the EDUs produced in the previous model. These probabilities are calculated based on structural and relational probabilities after all RST trees being converted into a set of tuples. Each tuple has the form R [i,m,j] that indicates a rhetorical relations between textual unit spanning over units i through m and the textual unit spanning over $m+1$ through j. Thereafter, the discourse parser model employs a set of features termed as dominance set to estimate the structure probabilities, and the discourse trees accordingly can be derived. The

dominance set contains features of syntactic and lexical information related to the point that links pair of EDUs. Generally speaking, the experimental results have surpassed the one obtained by Marcu (2000a). Furthermore, Soricut and Marcu have stated that SPADE would achieve accuracy that matches near-human levels of performance if it is provided with manual segmentations.

- ***Waleed Al-Sanie (2005):*** In his master thesis, Al- Sanie (2005) has presented the first attempt to automatically derive Arabic discourse structure using RST. His system infrastructure has been developed mainly for the task of Arabic text summarization. Al-Sanie (2005) has identified eleven rhetorical relations that are, in his view, suitable for the Arabic text. The nominated relations have been extracted by surveying all rhetorical relations formulated for the English language and selecting only the ones that comply with the rules set by the Arabic literature scholars. The identified relations along with their English equivalent are presented in Table 4-3. With respect to the parser, Al-Sanie has adopted the methodology introduced by Marcu (2000b). He has used cue phrases in order to break texts into EDUs; furthermore for each rhetorical relation he has assigned a set of these cue phrases that may indicate the presence of specific relation. Cue phrases have been associated with features so that the relations can be hypothesized based on their values. Eventually he has employed the 12 axioms proposed by Marcu (2000b) to generate all RST trees.

English Relation	اسم العلاقة	English Relation	اسم العلاقة
Condition	شرط	Result	نتيجة
Interpretation	تفسير	Example	تمثيل
Justification	تعليل	Base	قاعدة
Recalling	استدراك	Explanation	تفصيل
Confirmation	توكيد	Joint	عطف
Sequence	ترتيب		

Table 4-3: Arabic rhetorical relations identified by Al-Sanie.

However, no details have been given about the algorithm he has used to build up his sub-trees. His observations suggest that among all RS-trees the balanced ones appear to be the most suitable for the Arabic language rather than the most skewed to the right. This is due to the tendency of the Arabic writer to express his thoughts in a sequence of facts where each one is followed by statements to support it. The experiments in his dissertation have aimed at evaluating whether the textual fragments selected by his automatic summarizer are the most important units in that text.

- ***Daphne Theijssen (2008):*** The emergence of RST Treebank of annotated English texts has enabled researchers to develop models that employ machine learning algorithms; the study carried out by Theijssen (2008) is one example. In her study, Theijseen has assumed that sentences are the basic units of a text structure; subsequently her research has revolved around finding rhetorical relations between Multi sentential Discourse Units MSDUs within the same paragraph. In order to avoid complications of the RST parsing, Theijssen has restricted the scope of discourse analysis to the binary tree; she has also left out the directions and types of relations. To reach her goal, she has extracted triples (*x*-*y*-*z*) of three adjacent text spans located in the RST Treebank, where the span in the middle is either rhetorically related to the left or to the right span. The collected data consists of 2136 triples represent 942 different paragraphs. Thus with such training set, Theijssen has adopted five different learning algorithms with the aim of the automatic extraction of values for each of the potential relevant features. These features may lead to the detection of whether a text span is rhetorically related to the preceding or the following MSDU. She has investigated numerous features proposed by the previous studies in addition to examining 200 relations from the RST Treebank. The considered features have been split into five different categories that subsume: surface features, syntactic features, lexical features, reference features and discourse features. For accuracy measurements, she has used the relations that have been correctly selected by chance (56.0%) as a baseline, only the *Naïve Bayes* and *Maximum Entropy* machine learning algorithms have achieved an accuracy considerably better than the baseline with 60.0% and 60.9% respectively. Theijssen has stated that not being able to reach a good accuracy is due to the application of machine learning algorithms with their default settings, the small data set, and the large number of features.

- ***Vanessa Wei Feng and Graeme Hirst (2012):*** The RST parser developed by Feng and Hirst (2012) is another attempt of employing RST Treebanks at the full text level. Feng and Hirst have extended the HILDA discourse parser (Hernault et al., 2010) in which a variety of lexical and syntactic features have been extracted from input texts. Feng and Hirst (2012) have revised HILDA features set by incorporating various rich linguistic features into text-level discourse parsing, for example, semantic similarities, verb classes, cue phrases, production rules and contextual features that encode the discourse relations assigned by the preceding and the following text span pairs. Following the same methodology as in the HILDA parser, Feng and Hirst (2012) have used two classifiers for discourse tree building. The binary structure classifier to decide whether two consecutive text units should be merged to form a new sub-tree, and the multi-class classifier to evaluate which discourse relations are the most likely to hold between the new sub-tree. The parser performance has been measured under three discourse conditions: *Within-sentence*, *Cross-sentence* and *All level*. Their experimental results for the *Structure* classification task have achieved an F-score of 91.45, 55.87, and 89.51 under the three discourse conditions respectively. Whereas, the accuracy achieved for *Relation* classification task is 78.06, 46.83, and 65.30 under the same discourse conditions. Obviously, the parser performance is relatively poorer under the second discourse condition i.e. cross-sentences than that on within-sentence which, the authors have stated, indicates "*the difficulty of text-level discourse parsing*".

4.4 Discourse Markers

Discourse Markers (DMs) also known as *cue phrases*, *discourse connectives*, *coherence markers* and other names, draw mainly from the categories of conjunctions, prepositional and adverbials phrases. Interest in DMs has started with the shift in linguistics studies from focusing on the sentence as the higher unit of analysis into looking at the text as a whole (Al-Kohlani, 2010).

DMs have an important linking function that link adjacent segments (clauses, sentence, paragraphs) of discourse together to achieve coherence and cohesion. More importantly, DMs are frequently used by writers to avoid possible unintended interpretations of texts,

Al-Kohlani (2010) has stated "*This is an approach which views text as a communicative cohesive structure rather than a static one, and discourse markers as essential communicative tools that writers use to guide the reader's interpretation of their contribution in order to ensure a successful communicative act*".

4.4.1 Importance of Discourse Markers for NLP

A review of the major studies that have tackled the task of automatic discourse analysis, reveals that they share the assumption of considering lexical connectives – DMs – the most important type of signals in texts and their function is primarily to link linguistic units at any level, i.e. the main function of DMs are to structure the discourse.

One reason why DMs have been at the centre of the research on relation signalling is attributed to the fact that the distribution and frequency of DMs is sufficiently large to enable the derivation of rich rhetorical structures for texts, "*the number of discourse markers in a typical text is approximately one marker for every two clauses"* (Marcu, 2000b). Furthermore, numerous studies on discourse analysis have repeatedly shown that DMs are used frequently by writers to focus on the most important shifts in their narratives, mark intermediate breaks, and signal areas of topical continuity (Schneuwly, 1997; Sanders and Noordman, 2000). Therefore, it is likely that DMs can accelerate text comprehension, i.e., the occurrences of DMs, during reading tasks, leads to a faster processing of the subsequent text segment and recognition of a probe word.

One issue here is that DMs are considered as syntactically and semantically optional. However, a discourse that missed the presence of these linguistics units would be judged *disjointed*, *unnatural*, *impolite*, *unfriendly* or *awkward* within the communicative context (Brinton, 1996). The absence or underuse of DMs, therefore, may increase the chances of communicative breakdown (Al-Kohlani, 2010).

On the other hand a number of discourse analysts have argued that the effect of coherence markers depends on prior knowledge; readers who have less knowledge about the text topic, which is also the case of QA systems, are helped by these linguistic marking in establishing the relations that the author intend. In contrast, readers who are more familiar with the text content carry out better when reading a text without explicit markers (Kamalski et al., 2008;

McNamara and Kintsch 1996). All the above mentioned reasons make DMs the primary source of information for the tasks of automatically determining elementary units, hypothesizing relations between them and constructing rhetorical trees.

4.4.2 Discourse Markers as a Problematic Concept

Adopting a specific set of DMs is a challenging task, as a given word or expression may be classified as a DM by one researcher but not by another. This is due to the disagreement among researchers on the features and functions that exactly constitute a DM. These divergences reflect the different perspectives towards issues such as: the type of meaning they express, the semantic and syntactic features of these expressions and the role they serve in the text (Brinton, 1996).

A substantial number of studies have investigated the distinctive features and functions of DMs in a way to find out the characteristics and aspects that set them apart from other linguistic items. The fact that DMs do not have a unified grammatical status in addition to the variety of functions which they may operate at discourse level makes them a controversial issue. Therefore, each study has produced different descriptions of these functions, Al-Kohlani (2010) has indicated that, "*according to the way that discourse is viewed in each study and how it is approached*". She has also added another factor that has influence in determining the type of the functions "*The way in which the meaning of the items under investigation is perceived*".

With respect to the studies adopting discourse prospective approach, there is more than one view through which discourse can be seen, and accordingly different views of what constitutes a DM. One view of discourse which proposed by Schiffrin et.al (2001) incorporates such factors as structural, semantic, pragmatic cognitive and social in order to consider discourse *"as a process of social interaction"* thus, DMs would act *"in cognitive, expressive, social and textual domains"*.

Another issue that causes for disagreement among researchers is the status to be associated with DMs in terms of their meaning. For many researchers it is essential that a linguistic item being void of meaning in order to be classified as a DM, and accordingly any expression that

holds a conceptual meaning such as "*indeed*", "*frankly*" and "*next*" should be ruled out from considerations. Conceptual meaning may refer to *semantic, lexical, propositional, referential, or representation* content. In this regard, DMs are assumed to be lexically empty and confined to the pragmatic level such as "*in other words*", "*for example*" and "*as a result*". The issue here is that embracing the void of meaning status adds to the disagreement yet further, as the "non-conceptual" term implies different notions for different researchers. For example, while the expression "*in other words*" has been considered as non-conceptual DM by Fraser (1996), in contrast Blakemore has stated that this linguistic item is both nontruth-conditional and conceptual (Blakemore, 2003).

Researchers have also approached DMs from different points of view in terms of the multi-functionality characteristic. Some of them have considered that DMs have a unique function to serve in discourse. According to this view DMs should only denote one clear plane of meaning, since the multi-functions stance can lead to many interpretations by the reader. In contrast, other researchers have accepted the idea of pluralism pointing out that DMs may indicate more than one type of relation in the text at the same time. A stark example of this is the coordinating conjunctions that can play a discourse role in some instances i.e. they signal a rhetorical relation between two textual units while in other instances they play sentential or syntactic role, which adds to the ambiguity issue here.

Consequently, the conflicting views on identifying a general definition of DMs makes it impractical to adopt an exhaustive list. Therefore, it is essential for a scholar to perform a language-specific investigation and such a thing need to be conducted within the scope of the objectives of his study, as Lenk (1998) has reported "*It seems that every study of discourse markers must come up with its own definition depending on which items are being investigated in which type of discourse and within which framework*".

4.4.3 Arabic Discourse Markers

As we have mentioned above, DMs form a heterogeneous class of words and expressions drawn from different grammatical categories. There is no generally agreed list recognized by all researchers for the English language, and the Arabic is not an exception.

A number of scholars in linguistic literature have referred to the Arabic DMs broadly in the course of their research while discussing other language phenomena. Nevertheless, they have normally approached them from a syntactic perspective i.e. they have focused on the connective function of DMs by restricting their investigation to the sentence boundaries (Wright and Caspari 1896; Fareh and Hamdan, 1999).

However, only few studies went further and dedicated their work to the analysis of the role that DMs can play to tie units together at the discourse level. The works conducted by Sarig (1995) and Kammensjo (2010) are two examples of the attempts carried out to understand the use of this conceptual elements. While the former has examined DMs in "Contemporary Written Arabic" environment, the later has handled them in the spoken mode of Arabic language.

A more recent account of DMs has been proposed by Al-Kohlani (2010) who has presented a significant contribution to this area of research. She has provided a comprehensive description of the characteristics and features attributed to DMs and how these linguistic items operate at two levels of text structure (sentence and paragraph). Moreover she has conducted an extensive analysis on Arabic newspaper opinion articles in order to study the type, frequency and distribution of these devices. As a consequence she has identified a list of Arabic DMs used in opinion articles each of which is associated with a level of text (sentence or paragraph).

Al-Kohlani has applied the technique proposed by Kammensjo (2010). She has started by segmenting texts into paragraphs and sentences levels, then describing the coherent relations that relate textual units at each level and finally identifying groups of DMs classified according to their functional roles.

In order to achieve the goal of this chapter which is the automatic extraction of Arabic text structure, DMs are incorporated into our *Text Parser* model. This enables the *Text Parser* to acquire an appropriate representation of text structure relations. In this study, the *Text Parser* makes use of the DMs proposed by Al-Kohlani. However, out of her list we have only considered those associated with the sentence level as indicators of the presence of rhetorical relation between sentences. Appendix IV presents the DMs employed in the current study.

The main reason why the current study opts for Al-Kohlani's DMs is because she has employed two analytical tools to study the functional relations that relate textual units as coherent whole, namely the *Text-type Theory* and the *RST*. She has utilized the *Text-type Theory* to describe the relations that relate paragraphs to each other "*Global Relations*". On the other hand, she has used *RST* in discovering functional relations that occur between sentences "*Local Relations*". Figure 4-5 taken from Al-Kohlani (2010) illustrates this topology. As such, the outcome of her analysis should be consistent with the methodology adopted in this study since we employ the same framework i.e. extracting text structural organization based on RST. In what follows we shed some light on the characteristics and features of the environment in which she has conducted her data analysis.

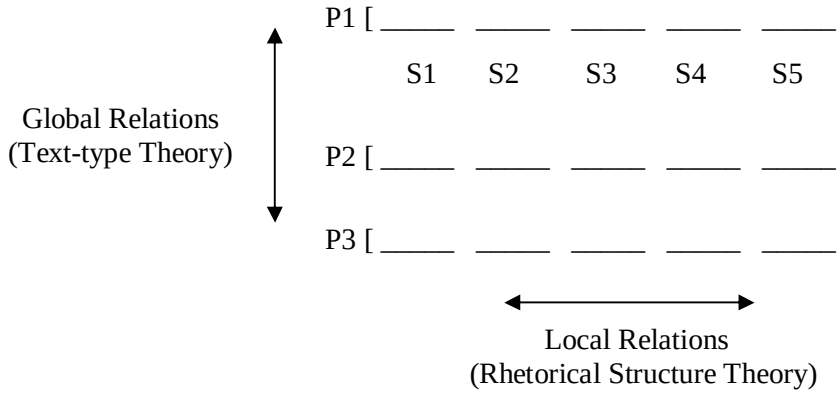

Figure 4-5: Text relations presented by Al-Kohlani (2010).

Text analysis has been conducted based on a corpus of around 30,000 words from 50 Arabic newspaper opinion articles. The articles have been extracted from the electronic editions of two international newspapers (*al-sahrq al-Awsat* and *al-Hayat*). Each article has been written by different professional Arab writers, with an average length of 900 words and has been geared towards native readers. Since the articles are of varying length, the articles set includes more than one article written by the same author, this relaxation has been allowed to equalize the size of material represented by each author. Al-Kohlani has restricted the length of the article to the 1500 words limit stating that "*long texts usually pose difficulties in following the argumentation points*".

Al-Kohlani has assumed that the collected articles would have "*an organizational plan*" because they have been produced by expert writers who have been in such profession for a

long time. Furthermore, it has been expected to yield a large number of DMs, most of which occur several times displaying a consistent pattern in their use. She has justified her selection of this genre of text for being "*simple, widely-used, and practical style that ventures to adopt new expressions and structures in order to be able to express the concepts of modern life*"

The issue here is that Al-Kohlani's work has been concentrated on specific genre prose i.e. newspapers opinion articles, consequently the produced DMs represent only one text-type which may result in different DMs set in case that a different one is investigated. What makes these DMs appropriate samples for our study, is the fact that this style of scripts is characterized as being of argumentative and evaluative nature that aim to influence readers' perceptions of facts and events. This implies that whenever writers seek to argue facts or express point of view, they tend to use the same DMs for such a purpose. Accordingly, employing these DMs is particularly useful for the objective of the present study.

4.5 The Construction of RS-Tree

This section presents the complete process by which the *Text Parser* model computes the complete formalization of a written Arabic text in order to automatically build up the plausible Tree representing the whole text based on RST.

Apparently, a system for automatic discourse analysis that creates full rhetorical structure in large-scale for Arabic text is currently unavailable. This is because of the high computational complexity involved in generating all valid RS-Trees resulting from processing a large number of hypothesized relations (Corston-Oliver, 1998; Marcu, 1997). Therefore, a more practical approach appears to be necessary to operate systems that are intended to locate answers to "*why*" and "*how to*" questions.

It is crucial to adopt an improved method that would be able to reduce the search space. This reduction can be achieved by decomposing the task of discourse structure derivation into two sub-tasks: detecting relations within sentences (intrasentential) and locating relations between sentences (intersentential). Obviously, considering relations spanning over only individual sentences one at a time is more computationally efficient than regarding the whole text. Furthermore, associating each hypothesized RST relation with a heuristic score would influence and guide the *Text Parser* to follow the track that would lead to construct the most

suitable tree rather than generating combinations of trees, thus avoiding any computational explosion.

In the current study, two models are incorporated to establish the proposed methodology. The first model, the *Pattern Recognizer* which segments text into EDUs sentences and provides semantic relations using the linguistic patterns formulated in Chapter 3. The second model, the *Text Parser* is built on top of sentences already associated with relational slots - provided by the first model - and aimed to posit rhetorical relations between adjacent textual spans consisting of at least one sentence. The *Text Parser* model has two main modules: the *Relation Recognizer* and the *Tree Builder* introduced in Sections 4.5.2 and 4.5.4.

4.5.1 Type of Texts

Texts are created with the aim of informing the reader about a specific subject. On his way to develop a text, the writer has to comply with some constraints as the reader is supposed to fully understand his text. A well-formed text must have sufficient signals of surface cohesion, for that is the best way for the author to avoid possible unintended interpretations.

In this study, two crucial assumptions underlie the process of automatically annotating text structure. The first is that the text is well-constructed i.e. cohesive and coherent. Cohesion across sentences has been investigated by Halliday and Hasan (1976). In their study, they have viewed the text as a unified whole in which the sentence is the highest unit of grammatical structure. Thus cohesion refers to "*the set of semantic resources for linking a sentence with what has gone before*" (Halliday and Hasan, 1976).

It is important to point out here that it is possible to write a cohesive text without necessarily being coherent. For example, the two sentences in text (22) embed the DM "لهذا" "*as a result of*" which indicates the presence of a reason relation, but the text cannot be perceived as coherent since it does not display any kind of logical order or consistency.

(22) معظم المدخنين مدمنين على السجائر. ولهذا فإن الأجسام النانوية أصغر بكثير من الأضداد وهي أيضا غير كارهة للماء كيمائيا.

"Many smokers are addicted to cigarettes. As a result, nano bodies are so much smaller than antibodies and are not chemical hydrophobic"

In this context, Reinhart (1980) has presented a description of coherence arguing that the text must meet three conditions in order to be coherent: Connectedness "*requires that the sentences of the text will be formally connected*", Consistency "*each sentences will be consistent with the previous sentence*", and Relevance "*is a pragmatic condition that restricts the relations between the sentences of the text and their context*".

The other assumption concerns the medium of the data that is being processed; the system developed in the current study deals with the Arabic text written in Modern Standard Arabic (MSA) form. This form is recognized by all Arab countries in addition to being the major medium of communication for public speaking and broadcasting and serious writing such as magazines, textbooks, newspapers, academic books and novels.

4.5.2 Recognizing Discourse Relations

As we have noted above, different techniques have been used in order to determine rhetorical relations. The *Relation Recognizer* proposed here adopts a rule based approach that relies on a set of heuristic scores. It takes the outcome of the *Patter Recognizer* model as input in the form of EDUs each of which is as long as a full sentence annotated with intrasentential relations, and it outputs a set of all possible rhetorical relations that may hold between these sentences. In most cases, a sentence is directly linked to the sentence that went before or to the sentence that comes after. In some cases, relations can be hypothesized between non adjacent sentences. Two types of relations can be posited, the first one that connects *nucleus* span with a *satellite* one is called *Hypotactic Relation*, whereas the second one which connects two nucleus spans is called *Paratactic Relation*.

4.5.2.1 Recognition of adjacent Relations

The *Relation Recognizer* first discovers rhetorical relations between adjacent sentences. It uses the linguistic devices that have been specifically gathered from the list of DMs generated by Al-Kohlani (2010). For example, a *Result* relation can be hypothesized between the two sentences in text (23) based on the occurrence of the expression "علیه" that appears at the head of sentence [2] as illustrated in Figure 4-6. The *Relation Recognizer* scores each of the identified relations according to its heuristic score that reflects its importance in building the text structure. Heuristic scores are discussed in Section 4.5.3.

(23) [قالت دراسة نشرت في صحيفة بريتش ميديكال إن الشاي الاسود الذي تم إعداده عند درجة حرارة تزيد عن 70 درجة مئوية يزيد من خطر الإصابة بالسرطان.][1] [و عليه يمكن تفسير ارتفاع الإصابة بسرطان المري بين بعض الشعوب الغير غربية.][2]

[The research published in the British Medical Journal found that black tea made at temperature greater than 70 c, can raise the risk of cancer,][1] [and that may be the cause of high rates of esophageal cancer among non western people.][2]

Figure 4-6: The schema of text (23).

On the other hand, in most cases the absence of DMs correlates with a preference to consider the statement in the unmarked sentence as continuation of the topic of the sentence that precedes it (Segal et al., 1991). Hence, there are two possible relations that can be hypothesized to hold between two unmarked sentences. One is an *Elaboration* relation when two sentences tackle the same point. The second relation is *Joint* which can be assumed to exist in case a topic shift occurs at the boundary between the two sentences.

Arab writers use demonstrative pronouns frequently to refer to the idea (question, proposition or event) which has been posed in preceding context (Zaki, 2011). In this regard, demonstrative pronouns which normally precede a noun made definite by prefixing the definite article play an important role as referring expressions. The demonstrative pronoun "هذه" that appears at the head of sentence [2] of text (24) illustrates this fact.

(24) [تفحص الطائرات بشكل دوري للتاكد فيما اذا كان هناك خلل في اي جزء من جسم الطائرة.][1] [هذه الاختبارات ضرورية لتجنب اي مشاكل محتملة.][2]

[The Aircraft is inspected regularly for any damage to any part of the fuselage.[1]] [***These*** checks are crucial in order to avoid any potential problem.[2]]

However, demonstrative pronouns also used to refer to some other entities which appear in the same sentence. Consider for example text (25) in which the pronoun "هذا" refers to the idea stated at the beginning of the sentence. This effect can be attributed to the position of the pronoun as it is located approximately in the middle of the sentence. During our experiments, we have observed that whenever a demonstrative pronoun occurs within a window comprising the first third of a sentence it most likely refers to an entity located in the previous sentence; and the second sentence accordingly is considered to elaborate on the first one. Table 4-4 presents the set of demonstrative pronouns employed in this study.

(25) من المرجح ان تزيد كمية المعطيات المتاحة للتحليل والتقيمم بشكل كبير مع مرور الوقت وهذا الامر يعني فسح المجال لفرص عمل تتطلب التفكير بطريقة حاسوبية من اجل ترتيب المعلومات وجعلها قابلة للاستخدام.

The amount of data available for evaluation and analysis is likely to increase drastically with the passage of time and ***this*** means an opening of job opportunities that require computational thinking in order to sort out the information and make it usable.

	Proximal	Distal
Singular	هذا - هذه	تلك – ذلك
Dual	هؤلاء	اولئك
Plural	هاتان – هذان – هاتين – هذين	

Table 4-4: Demonstrative pronouns forms in Arabic

After all rhetorical relations have been hypothesized, a *Joint* relation is applied to connect all adjacent sentences that no actual relation has been found to relate them. This point is discussed in more detail in Section 4.5.4.3.

4.5.2.2 Recognition of distance Relations

Given our commitment to the assumption we have made in Section 4.5.1 i.e. the text to be derived is well-constructed, it is possible that one sentence in the middle of the text might be related to another in the beginning.

In his well known work, Marcu (2000b) has associated each DM with the feature "*Maximal distance*" which specifies the number of sentences that separates the textual units that are related by that DM. In case a marker has been assigned the value -1, the two related sentences are adjacent. This value has been determined based on corpus analysis. For example, the marker *Although* has been given the value 5 when trying to signal *Elaboration* relation i.e., the relation *Elaboration* is hypothesized to relate the sentence that contains this marker with the sentence that directly precedes it, and also relates the sentence with the sentence that comes before and so on within a maximum distance of 5.

However, the outcome of this approach comes at the cost of computational complexity, as the number of hypothesized relations increases, the number of sub-trees increases exponentially. The text (26) taken from (Marcu, 2000b) illustrates this situation in which the occurrence of the DM *In contrast* contributes to make the following exclusively disjunctive hypothesis *rhet_rel* (CONTRAST, A, C) $\oplus$ *rhet_rel* (CONTRAST, A, D) $\oplus$ *rhet_rel* (CONTRAST, B, C) $\oplus$ *rhet_rel* (CONTRAST, B, D). Moreover, associating each marker with a fixed number of textual units may result in inappropriate relations especially when positing relations at the sentence level; as the number of sentence is highly related to the context of the text in which such marker appears.

(26) [John likes sweets.^{A}] [Most of all, John likes ice cream and chocolate.^{B}] [*In contrast,* Mary likes fruits.^{C}] [Especially bananas and strawberries.^{D}]

Croston-Oliver (1998) has used a different method which checks all pairs of clauses in a text in an effort to hypothesize all possible discourse relations. These hypothesized relations are then grouped into bags of mutually exclusive relations i.e. one and only one of the possible relations belongs to the same bag. Nevertheless, for large texts, the time complexity for examining the constraints corresponding to all possible relations could be also high.

An attempt for annotating this sort of relation has been introduced by Mathkour, Touir and Al-Sanea (2008) in their work on Arabic text summarization. According to their observation,

there is an implicit transitivity relation over hypotactic relations. The sentences in text (27) demonstrate this fact. We notice that sentence [2] elaborates the idea mentioned in sentence [1]; also, the DM "لذلك" "Therefore" signals a rhetorical relation of *Result* between sentences [3] and [2]. However, the information stated in sentence [3] is still considered as a result of the idea presented in sentence [1]. Hence, according to the transitivity principle we can say that a hypotactic relation of *Result* also holds between sentences [3] and [1]. The schema in Figure 4-7 shows the discourse analysis of text (27).

(27) [يعتقد خبراء التجميل أن البشرة الصافية والخالية من البقع وحب الشباب والتجاعيد هي من مقومات الجمال.][1] [حتى ذهب البعض إلى القول "لا جمال بدون بشرة جميلة".][2] [لذلك نرى اهتمام الجميع بالمحافظة على بشرة جميلة.][3]

[Beauty experts believe that one of the fundamentals of beauty is to have a skin that is free of spots, acne and wrinkles.][1] [Some **even** went as far as saying: "there is no beauty without a beautiful skin".][2] [**Therefore** everybody is keen about having a beautiful skin.][3]

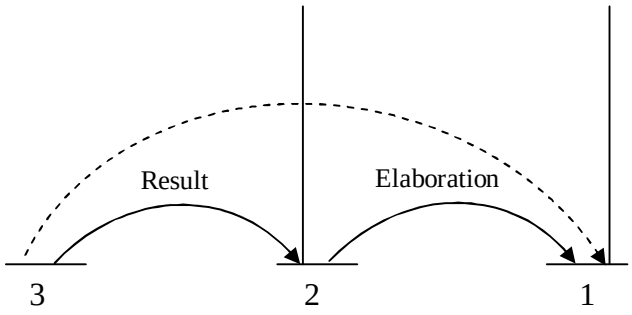

Figure 4-7: A rhetorical analysis of text (27).

A different approach for discovering distance relations among sentences has been utilized by Timmermn (2007), in which the keyword repetition has been used as indicator of the presence of a distance relation. The idea behind this technique relies on a facet of text coherence that is adequate for determining the sentences that have a single theme i.e. if two sentences tackle the same point it is likely that they involve the same elements of nouns. In this sense, we can say that a *Hypotactic Relation* relates those two sentences. In fact, it is difficult to accurately recognize which type of relation exists without world knowledge. However, in this study, the added relation will always be considered as an *Elaboration* relation where the sentence that comes after (*satellite*) elaborates on the topic of the sentence that came before (*nucleus*).

The matching process is carried out as follows: the words that are associated with noun tags are initially extracted and then all suffixes of these nouns are removed using a word stemmer. Thereafter, each sentence is compared to the following sentences in turn. If a match is found, a new relation is hypothesized to hold between the two sentences under consideration provided that neither sentence is rhetorically related to another one; this condition is neglected in case that the sentence has the *nucleus* status.

Let us consider the four sentences in text (28), we notice that a rhetorical relation of *Result* is signalled between sentences [1] and [2] based on the occurrence of the DM "مما أدى" at the head of sentence [2]. Since sentence [1] is the *nucleus* of this relation, it is matched with sentences [3] and [4] for possible mutual nouns. Finally two hypotheses of *Elaboration* relations are added to the relations set because the sentences share the nouns "ارض - نيزك" "meteorite - Earth". The schema in Figure 4-8 shows the discourse analysis of text (28). In the current study we adopt the transitivity method and the repetition of keywords in order to recognize long distance relations.

(28) [أكد بعض العلماء أن نيزكا كبيرا اصطدم بالأرض في حقبة الديناصورات منذ ملايين السنين.][1] [**مما أدى** الى هلاك هذه الديناصورات والأحياء الأخرى التي عاشت في تلك الفترة.][2] [وتم التعرف على آثار النيزك من خلال طبقة الرواسب المتخلفة عن السحابة الغبارية التي غطت كوكب الأرض بعد الاصدام.][3] [ان دراسة اصدام النيزك بالكرة الارضية يمكن ايضا ان تساعد العلماء على فهم الظروف التي نشأت فيها الحياة على هذا الكوكب بشكل افضل.][4]

[A team of researchers has confirmed that a large meteorite had collided with Earth at the age of dinosaurs millions of years ago.][1] [This was **responsible for** the mass extinction of dinosaurs and all other species living on Earth.][2] [The meteorite was identified from the layer of sediment deposited from the dust cloud that enveloped the Earth after the impact.][3] [Studying the meteorite's impact with the Earth could also help researchers better understand the conditions under which early life on the planet evolved.][4]

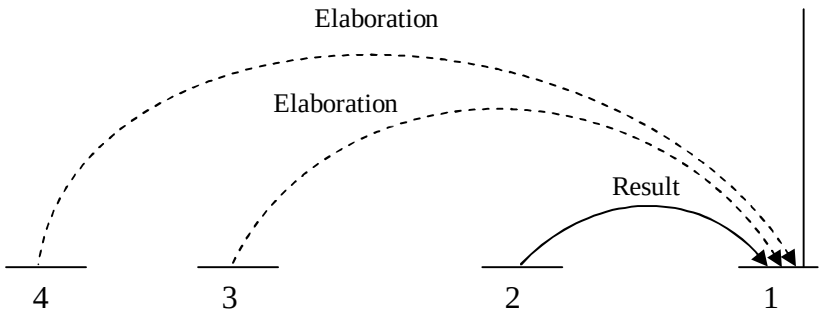

Figure 4-8: A rhetorical analysis of text (28).

4.5.3 Heuristic Scores

In employing the DMs proposed by Al-Kohlani (2010), it is important to emphasize that we have not embraced all the rhetorical relations she has presented. Rather, a set of ten relations have been adopted in this study. In fact, these relations occur more often among sentences and represent relations that are sufficient for reflecting writer's attitudes and viewpoints in discourse from cohesion-based perspective. The other relations by Al-Kohlani are hardly signalled in text. Table 4-5 shows the adopted relations.

Reason	Background
Interpretation	Certainty
Evaluation	Contrast
Result	View
Sequence	Elaboration

Table 4-5: List of the rhetorical relations employed.

Each possible relation receives a heuristic score that reflects its relative importance throughout the automatic text derivation process. Since the main aim of the current study has been set to provide answers for "*why"* and "*how to"* questions, rhetorical relations which are more relevant for such type of questions should be highlighted. Thus, we have chosen a small subset of the ten rhetorical relations adopted. The concerned subset consists of the following relations: *Result*, *Reason*, and *Interpretation*. Our goal then is to prioritize the relevant relations subset in order to ensure that its members are always in the sub-trees produced by the *Text Parser*. This can be achieved by assigning higher scores to this subset as discussed below.

One challenge of using DMs when discovering relations between sentences is that certain DMs are multi-functional i.e. they can signal more than one type of rhetorical relation in discourse. For example, the expression "من هنا" in sentence [A] of text (29) indicates a *Result* relation, whereas it implies an *Evaluation* relation that holds between sentences [A] and [B] of text (30).

(29) [يستطيع العلماء في الوقت الحالي متابعة النيازك في حجم كيلومتر أو أكبر.][A] [من هنا قام فريق عمل انجليزي بتوجيه تليسكوب دقيق في الجزء الجنوبي من الكرة الأرضية بهدف تحديد الأجسام الأصغر حجما.][B]

[Nowadays, scientists can track meteorites of a kilometre size or more.][A] [**Therefore**, an English working group has undertaken to direct a high precision telescope in the southern hemisphere in order to indentify smaller objects.][B]

(30) [إن أشعة الشمس ما بين الساعة الثامنة صباحا والساعة الخامسة مساء تؤدي إلى تنشيط الخلايا المولدة للصباغ وبالتالي تشكل البقع والكلف.][A] [من هنا فان الوقت المفضل للتعرض للشمس هو عندما يكون خيال الإنسان اطول من طوله.][B]

[Sunrays between 8 in the morning and 5 in the afternoon energise cells responsible for pigment and consequently forms spots and freckles.][A] [It **can be concluded** that the best time to be exposed to the sun is when the person's shadow is longer than him.][B]

Another indicator is ought to deal with this problem and avoid any kind of ambiguity. It may very well be the case that knowledge about the sentence structure containing that DM can be exploited. Let us consider text (29) again; we notice that sentence [2] includes an intrasentential *Causal* relation. This relation can be acquired using linguistic pattern P (11) which has been constructed using the *Pattern Recognizer* introduced in Chapter 3. Hence, the existence of cause-effect information increases the probability for an ambiguous DM to indicate one of the rhetorical relations belong to the relevant relations subset.

P (11) R &(C) [C2] (AND) (&This) بهدف/بغرض [C1] &.

Annotated corpora ought to be available to automatically learn the optimal values for heuristic scores. Unfortunately, no corpus of Arabic RST-analyzed texts exists. Hand-tuning is therefore still necessary. The heuristic scores presented in this study have been obtained by trial and modification with the aim of ensuring that preferred relations occurred at the top of sub-trees list. For example, *Result*, *Reason* and *Interpretation* relations are extremely good indicators of "*why*" and "*how to*" questions. We can therefore assign a high initial value, whereas *Elaboration* and *Background* relations are weaker indicators. We have carried out a regression test on Arabic texts and the outcome of the *Text Parser* is always checked to determine whether it produces a tree that spans over the whole text. Heuristic scores are then adjusted until *Text Parser* produces preferred analyses. Table 4-6 shows the maximum values of each relation.

Relation type	Maximum Score
Result	100
Reason	100
Interpretation	100
Elaboration	80
Contrast	70
Background	60
Evaluation	50
Certainty	50
Sequence	50
View	50

Table 4-6: Score assigned for each relation.

We have examined each DM in the list and considered its potential contribution in hypothesizing the rhetorical relations. In case a DM correlates with only one particular relation, that relation thus is indicated with a relatively high level of confidence and accordingly the DM has been associated a score that is equal to the maximum value of that relation. Whereas if a DM signals different discourse relations such as the DM "من هنا", it is perceived as a weaker evidence and accordingly it has been associated a low score. Table 4-7 shows a set of scores that correspond to some of the DMs.

Marker	Rhetorical relation	Score
من هنا	Evaluation – Result	50 – 40
من اجل ذلك	Result	100
إلا ان	Contrast	70
خصوصا ان	Reason	100
كما ان	Elaboration	80
بالتالي	Evaluation – Result	50 – 40
ثم	Sequence	50

Table 4-7: A list of DMs and corresponding heuristic score.

With regard to recognizing relations based on the facet of text cohesion, scores are calculated based on the number of similar keywords that co-occur in both sentences. If this similarity is above certain threshold, an *Elaboration* relation is considered to hold between the two sentences. The assigned score resides between 0 and 80 where each shared keyword adds the value 15. Also, the occurrence of a demonstrative pronoun adds a value of 60 to the accumulated score in the case of inspecting adjacent sentences. Finally, sentences are examined for the presences of intrasentential relations which add the value 45 to any of the relations in the relevant set. Table 4-8 shows a set of values that may be added by some indicators.

Type	Relation	Score
Shared noun	Elaboration	+15
Intrasentential relation	Relevant subset	+45
Demonstrative pronoun	Elaboration	+60

Table 4-8: The added scores for some types of indicators.

ALGORITHM 4-1 finds possible relations for a given text. The input constitutes a list of EDUs each of which is a complete sentence annotated with intrasentential relations. The *Relation Recognizer* operates from the bottom up. First, every pair of the adjacent sentences in the EDUs is checked for possible relations on the basis of DMs occurrences. Thereafter, the list is examined again for the presence of long-distance relations among sentences that have not been already hypothesized to be related to another EDU as a *satellite* unit. The *Relation Recognizer* employs heuristics scores to add a scoring value for each hypothesized discourse relation.

All generated relations are stored in an ordered set according to their heuristic score. In case that more than one relation is found to connect the same two sentences, the relation with the highest heuristic score is retained and all the others are discarded. At this point all sentences are supposed to be connected as the text is presumed to be coherent.

ALGORITHM 4-1: Hypothesizing rhetorical relations.

```
Input: A sequence S[n] of sentences annotated with
       intrasentential relations.
Output: A list RR of relations that hold among sentences in
        S[n].
1. RR:= null;
2. Determine the set DMs of all Discourse Markers occur at the
   head of each sentence in S[n];
3. For each marker M ∈ DMs
4.    rr:= null;
5.    While there is a relation that M can relate
6.     rr: = rr ⊕ rhet_rel(name(M), score(M), l(M), r(M));
7.    RR: = RR ∪ {rr};
8. For each pair (i,j) of adjacent sentences in S[n]
9.    If more than one relation found in RR to hold between(i,j)
10.        rr: = rr ∪ rhet_rel(name, score(max), i, j);
11. RR: = RR ∩ {rr};
12. For each pair (x,z) of sentences in S[n]
13.   Use cohesion and transitivity to find distance relation rrd
14.    If Score(rrd) > threshold
15.       RR: = RR ∪ rrd
16. Sort RR from the highest scored hypothesis to the lowest
    scored
```

4.5.4 Constructing Sub-Trees

Given a text segmented into EDUs at the sentence level and a set of rhetorical relations that have been hypothesized to hold between those sentences, we are now building up the possible RST Tree for that text. The *Tree Builder* applies the posited discourse relations with high heuristic scores before those with lower heuristic scores in a bottom-up manner, grouping contiguous clauses into a hierarchical representation.

4.5.4.1 Compositionality

Marcu (2000a) has proposed a compositionality principle to join two adjacent sub-trees: "*whenever two large text spans are connected through a rhetorical relation, that rhetorical relation holds also between the most important parts of the constituent spans*". This principle can be explained by text (31) taken from Marcu (2000b).

(31) [No matter how much one wants to stay a non-smoker, [A]] [The truth is that the pressure to smoke in junior high is greater than it will be any other time of one's life. [B]] [We know that 3,000 teens start smoking each day, [C]] [although it is a fact that 90% of them once thought that smoking was something they'd never do. [D]]

Applying RST to text (31) yields the set of relations shown in Figure 4-9. The task now is to construct RS-tree for text (31). Assume that one takes the decision to build the spans [A,B] and [C,D], as illustrated in Figure 4-10. To complete the construction of the discourse tree, a decision has to be made about the best relation that could span over [A,B] and [C,D]. Considering elementary rhetorical relations in Figure 4-9 that hold across the two spans, there are three choices: *rhet_rel* (JUSTIFICATION$_1$,D,B), *rhet_rel* (EVIDENCE,C,B), and *rhet_rel* (RESTATEMENT,D,A).

One can notice that the *Evidence* relation would be the best one because it is consistent with the compositionality principle i.e. the *Evidence* relation that holds between text spans [C,D] and [A,B] is explained by an *Evidence* relation that holds between their most important subspans (C and B).

$$RR = \begin{cases} rhet\_rel\ (\text{JUSTIFICATION}_0, A, B) \\ rhet\_rel\ (\text{JUSTIFICATION}_1, D, B) \\ rhet\_rel\ (\text{EVIDENCE}, C, B) \\ rhet\_rel\ (\text{CONCESSION}, D, C) \\ rhet\_rel\ (\text{RESTATEMENT}, D, A) \end{cases}$$

Figure 4-9: A set of possible rhetorical relations of text (31).

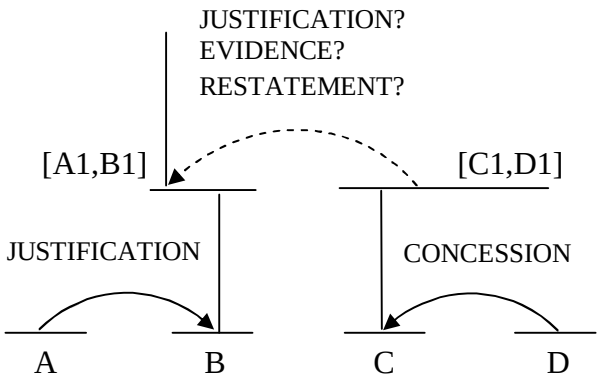

Figure 4-10: A rhetorical analysis of text (31).

Accordingly, Marcu has associated each rhetorical relation with a promotion set in order to reflect the compositionality criterion. Promotion set is the set of units that constitute the most important parts of the text that is spanned by the node. For a terminal node, the promotion set consists only of the terminal node itself. For an asymmetric sub-tree, the promotion set consists of a single element, the *nucleus*. For a symmetric sub-tree, the promotion set consists of the union of the promotion sets of the co-nuclei.

In this study, we assume full conformity to the principle of compositionality, which contributes to the production of well-formed tree and drastically reduces the size of the solution space.

4.5.4.2 Text Structure Formalization

The approach we take in formalizing rhetorical relations draws heavily on Marcu's work (Marcu, 2000b) in which he has given a clear description of an instance of a text structure. However, we have amended the formalization so that it includes the score feature introduced in Section 4.5.3.

The formalization uses the following predicates.

- Predicate *Position* (S_i, j) is true for a sentence S_i in sequence S if and only if S_i is the j^{th} element in the sequence.
- Predicate *rhet_rel* (*name*, *score*, S_i, S_j) is true for sentences S_i and S_j with respect to rhetorical relation *name* if and only if the rhetorical relation *name* and the *score* value are consistent with the relation between sentences S_i and S_j.
- Predicate *rhet_rel* (*name*, *score*, *S1s*, *S1e*, *S2s*, *S2e*) is true for textual spans [*S1s*, *S1e*] and [*S2s*, *S2e*] with respect to rhetorical relation *name* if and only if the rhetorical relation *name* and the *score* value are consistent with the relation between the textual spans that ranges over sentences *S1s*- *S1e* and sentences *S2s*- *S2e*.

A representation of the rhetorical relations found in text (29) is given in Figure 4-11.

rhet_rel (Result, 85, A, B)
rhet_rel (Evaluation, 50, A, B)
position (A, 1)
position (B, 2)

Figure 4-11: Representation of rhetorical relation of text (29).

Tree-based structures seem to be adequate representations of any text. Marcu (2000b) has stated *"Most discourse and text theories mention explicitly or implicitly that trees are good mathematical abstractions"*, he has added *"tree-based structures are also easier to formalize and derive automatically"*. As such, the following features constitute the foundation on which the formalization has been built:

- A text tree is a binary tree whose leaves denote elementary sentences.
- Each node has an associated *Status* (*nucleus* or *satellite*), a *Type* (the rhetorical relation that holds between the text spans that the node spans over), a *Promotion* (the set of units that are most important), and a *Score* (the value that reflects its priority).

Figure 4-12 illustrates an example of these features that correspond to the relation relating the two sentences of text (23).

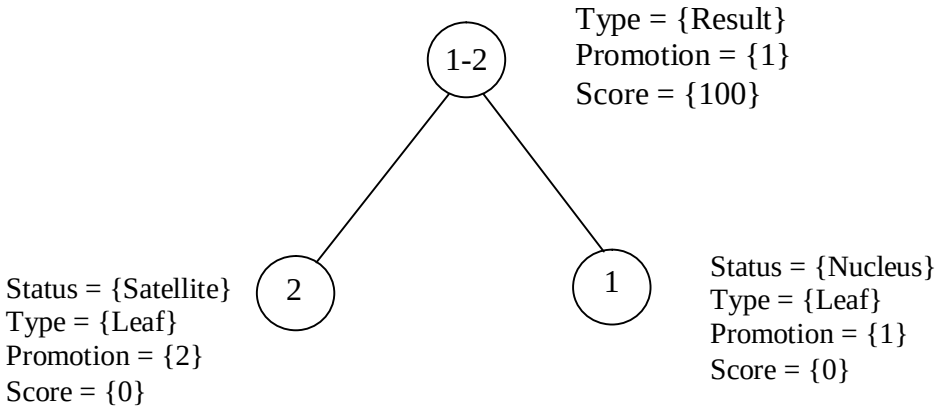

Figure 4-12: Features of the relation connecting the two sentences of text (23).

4.5.4.3 Building the RS-Tree

ALGORITHM 4-2 gives a description of the steps the rhetorical *Tree Builder* follows when it builds up the valid tree structure compatible with the set of hypotheses produced by the *Relation Recognizer*.

The *Tree Builder* establishes a list of sub-trees by gathering text spans into contiguous new textual units in accordance with the principle of compositionality which guarantees that only adjacent spans of text can be put in relation within an RST tree.

Each sub-tree takes the following form:

`SubTree(L,R,Status,Type,Promotion,Score,left_SubTree,right_SubTree)`

where [L ,R] are the left and the right boundaries of a sub-tree.

Sub-trees are being building up by iterating over all pairs in the relations set. The *Tree Builder* starts by selecting the relations ranked highest according to their scores since they constitute the most promising path, and then it moves to the second pair in the relations set. Heuristic scores are being accumulated by adding up all scores in the sub-trees constructed so far. This step is repeated until the list of sub-trees contains only one tree spanning over all sentences in the text. If no relations are found between two adjacent sub-trees, the sub-trees could be assembled with a *Joint* relation because in a well-written text no textual unit is completely isolated. In practice these inspections can be performed at very little computational expense.

ALGORITHM 4-2: Building up the valid tree structure.

```
Input: A text T of N sentences S[N]
       A sorted list RR of relations that hold among the
         sentences in S[N].
Output: The RS-tree of T.
1. SubTreesList := Null;
2. For i= 1 to N
3.    Convert sentence into the form
      SubTree(i, i, NONE, LEAF, {Si}, 0, NULL, NULL);
4.    SubTreesList:= SubTreesList ∪ SubTree;
5. End For
6. While RR contains at least one element and the SubTreesList
   has more than one element
7.   For each rr ∈ RR
8.      Search in SubTreesList for elements with the
         promotions specified by rr;
9.        If match not found or combining the two subTrees
          would result in crossing lines
10.          Remove rr;
11.       Else create a new subTree by joining the tow
          subTrees as specified by rr and add the heuristic
          score accordingly;
12.       Update SubTreesList and RR accordingly;
13. End While
14. If SubTreesList has more than one element
15.    Join all elements in SubTreesList into one tree that
       spans the whole text;
```

4.6 Worked Example

The operation of the text derivation developed in this study is illustrated by the example below. The example examines several processes executed by the *Pattern Recognizer* and *Text Parser* models by means of text (32).

(32) *تتعرض عظام المرأة والرجل بعد سن الخمسين إلى التنخر الداخلي وتناقص مستوى الهرمون المساعد على بنائها. بالتالي تكون العظام ضعيفة أمام الضربات والحوادث مما يؤدي إلى حدوث الكسور التي تتطلب أحيانا الاستبدال العظمي. والادوية الموجودة حاليا تعمل على تقليل نسبة الهدم في العظام من خلال المحافظة على نسبة الكالسيوم الطبيعية في الدم. لكن الدواء الجديد ويدعى فورتيو ، فهو على العكس يعمل على زيادة سرعة عملية البناء والتي تقوم بها الخلايا البانية للعظم. وفي التجارب التي اجريت على هذا الدواء طهر ان احتمالات اصابة العمود الفقري بالكسور قلت بنسبة 65- 90% بينما قلت احتمالات اصابة الكاحل والمعصم و الورك والاضلاع والقدم بالكسور بنسبة 54%*

After the age of fifty, bones of men and women are exposed to internal necrosis and reduction in the level of the hormone that helps building bone structure. As a result, and in cases of blows and accidents, bones become weak and prone to fractures which sometimes require bone replacement. Nowadays, the existing drugs reduce the ratio of bone destruction by maintaining the ratio of natural calcium in blood. On the contrary, the new drug, Forteo, accelerates the speed of the construction process carried out by bone-constructing cells. The experiments conducted on this drug revealed that the possibilities of spine fracture injuries decreased by 65% -90%, whilst possibilities of ankle, wrist, hip, ribs and foot fracture injuries are reduced by 54%.

The *Pattern Recognizer* starts with the segmentation process through which the text is split into elementary discourse units each of which with the length of a full sentence. This implies searching for the dot symbol in text. However, not every single occurrence of the dot is considered as a boundary segment. There are cases that require special attention, for example, abbreviation with dots followed by a proper noun should be excluded from the segmentation process.

Then the *POS Tagger* assigns a syntactical category for each token in the sentences. The segments of the text along with the POS tags obtained from *Stanford Tagger* are shown in Figure 4-13. The POS tags contain some errors; in sentence (E) for example, the word "ان" ought to be tagged as a particle *RP*. Also, the tag of the word "لكن" in sentence (D) is not correct.

The next step is to apply the linguistic patterns to discover *Causal* and *Explanatory* relations within sentences. This process yields *Causal* relation (33) from sentence (B) and *Explanatory* relation (34) from sentence (C) as seen in Figure 4-13.

[تتعرض/VBP عظام/NN المرأة/DTNN و/CC الرجل/DTNN بعد/NN سن/NN الخمسين/CD الى/IN التنخر/DTNN الداخلي/DTJJ و/CC تناقص/NN مستوى/NN الهرمون/DTNNS المساعد/DTJJ على/IN بنائها/NN .PUNC/][A بالتالي/JJ تكون/VBP العظام/DTNN ضعيفة/JJ امام/NN الضربات/DTNNS و/CC الحوادث/DTNN مما/NN يؤدي/VBP الى/IN حدوث/NN الكسور/DTNN التي/WP تتطلب/VBP احيانا/NN الاستبدال/DTNN العظمي/DTJJ .PUNC/][B و/CC الادوية/DTNN الموجودة/DTJJ حاليا/JJ تعمل/VBP على/IN تقليل/NN نسبة/NN الهدم/DTNN في/IN العظام/DTNN من/IN خلال/NN المحافظة/DTNN على/IN نسبة/NN الكالسيوم/DTNN الطبيعية/DTJJ في/IN الدم/DTNN .PUNC/][C لكن/VBP الدواء/DTNN الجديد/DTJJ و/CC يدعى/VBN فورتيو/NNP ،/PUNC فهو/NNP على/IN العكس/DTNN يعمل/VBP على/IN زيادة/NN سرعة/NN عملية/NN البناء/DTNN و/CC التي/WP تقوم/VBP بها/NN الخلايا/DTNN البانية/DTJJ للعظم/NNP .PUNC/][D و/CC في/IN التجارب/DTNN التي/WP اجريت/VBN على/IN هذا/DT الدواء/DTNN ظهر/NN ان/IN احتمالات/NNS اصابة/NN العمود/DTNN الفقري/DTJJ بالكسور/NN قلت/VBD بنسبة/NN CD/65 -/PUNC CD/90 %/CD بينما/IN قلت/VBD احتمالات/NNS اصابة/NN الكاحل/DTNN و /CCالمعصم/DTNN و/CC الورك/DTNN و/CC الاضلاع /DTNN و/CC القدم /DTNN بالكسور/NNP بنسبة/NNP CD/%54 .PUNC/][E

Figure 4-13: POS tags and segments of text (32).

The two relations below show cause-effect and method-effect parts that were extracted from sentence (B) and (C) respectively.

(33) [E] [تكون العظام ضعيفة أمام الضربات والحوادث] [C] [حدوث الكسور التي تتطلب أحيانا الاستبدال العظمي]

(34) [M] [الأدوية الموجودة حاليا تعمل على تقليل نسبة الهدم في العظام][E] [المحافظة على نسبة الكالسيوم الطبيعية في الدم]

Given sentences tagged with intrasentential relations, the *Text Parser* model then starts to identify rhetorical relations between these sentences. The *Relation Recognizer* first examines all pairs of the adjacent sentences and produces the hypothesized discourse relations given in Figure 4-14.

rhet_rel (Result, 85, A, B)
rhet_rel (Evaluation, 50, A, B)
rhet_rel (Contrast, 70, C, D)
rhet_rel (Elaboration, 60, D, E)

Figure 4-14: Adjacent relations for text (34).

We notice that two relations, *Result* and *Evaluation*, are posited between sentences (A) and (B) based on the occurrence of the DM "بالتالي" at the head of sentence (B). The score of the *Result* relation is calculated by adding 45 points to the base value 40 because sentence (B) is

tagged with *Causal* relation (33). The relation with the higher likelihood between sentences (A) and (B) is kept and the other one is discarded i.e. the *Evaluation* relation. Also, an *Elaboration* relation is hypothesized between sentences (D) and (E) based on the occurrence of the demonstrative pronoun "هذا" in the first third of sentence (E).

The *Relation Recognizer* proceeds with discovering long distance relations. It compares nouns in each possible pair of sentences and assigns a likelihood based on the number of similar nouns. The *Relation Recognizer* only adds an *Elaboration* relation if it receives a score above the threshold. For example, only the noun "عظام" is shared between sentences (A) and (C), thus such relation is not added to the relations list. Also, sentences (D) and (E) contain the noun "الدواء" which indicates the presences of an *Elaboration* relation with a likelihood of 15. However, since an *Elaboration* relation has been hypothesized between the same sentences in the previous step this value is added up the total score. At this stage all sentences are connected and the final relation set is shown in Figure 4-15.

rhet_rel (Result, 85, A, B)
rhet_rel (Contrast, 70, C, D)
rhet_rel (Elaboration, 75, D, E)

Figure 4-15: Relations set for text (32).

Next, the *Tree Builder* parses the relations list generated by the *Relation Recognizer*. It initially converts all sentences into terminal nodes represented as sub-trees each has a single member in its promotion set - the sentence itself. The *Tree Builder* then attempts to apply all the rhetorical relations starting with the one which has the highest score. Figure 4-16 illustrates the sub-trees list content resulting from the application of the first and third hypothesis in the relations set, sentences written in curly braces specify the promotion set of each sub-tree. The *Tree Builder* moves on to consider the *Contrast* relation, it searches the sub-trees list for a sub-tree whose promotion set includes sentence (C) and a sub-tree whose promotion set includes sentence (D). It finds the terminal node (C) and the sub-tree [D-E], it thus combines them to form a new sub-tree covering sentences (C) through (E) as shown in Figure 4-17. The *Tree Builder* is unable to find a relation that connects sub-tree [A-B] and [C-E], and therefore a *Joint* relation is applied to combine the two sub-trees. Figure 4-18 depicts the Tree that covers the entire input text.

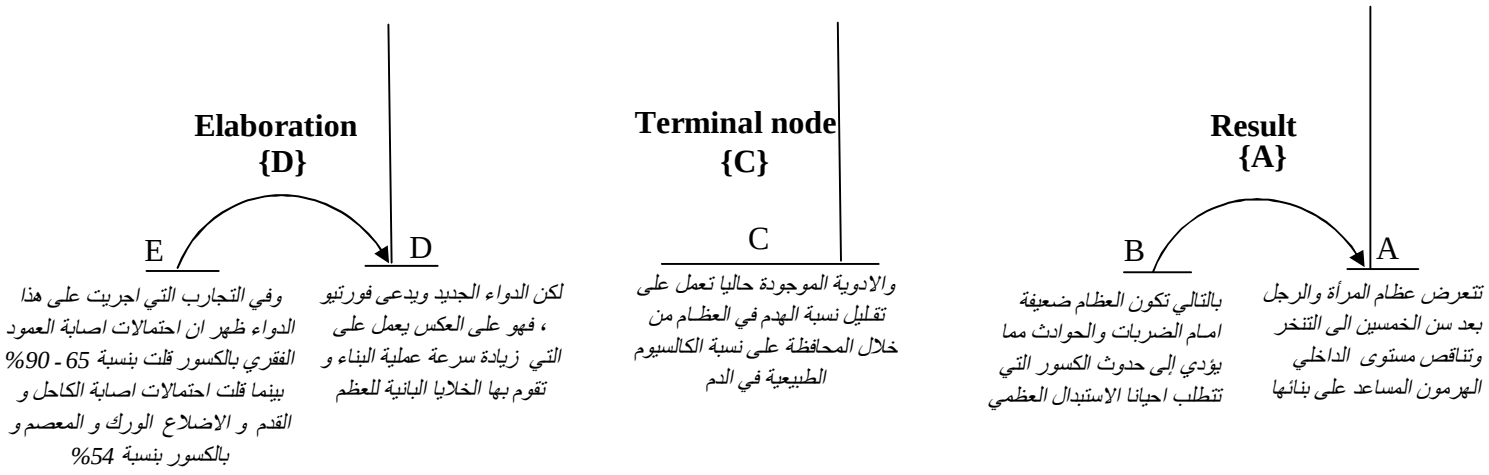

Figure 4-16: Sub-trees list after applying the *Result* and *Elaboration* relations.

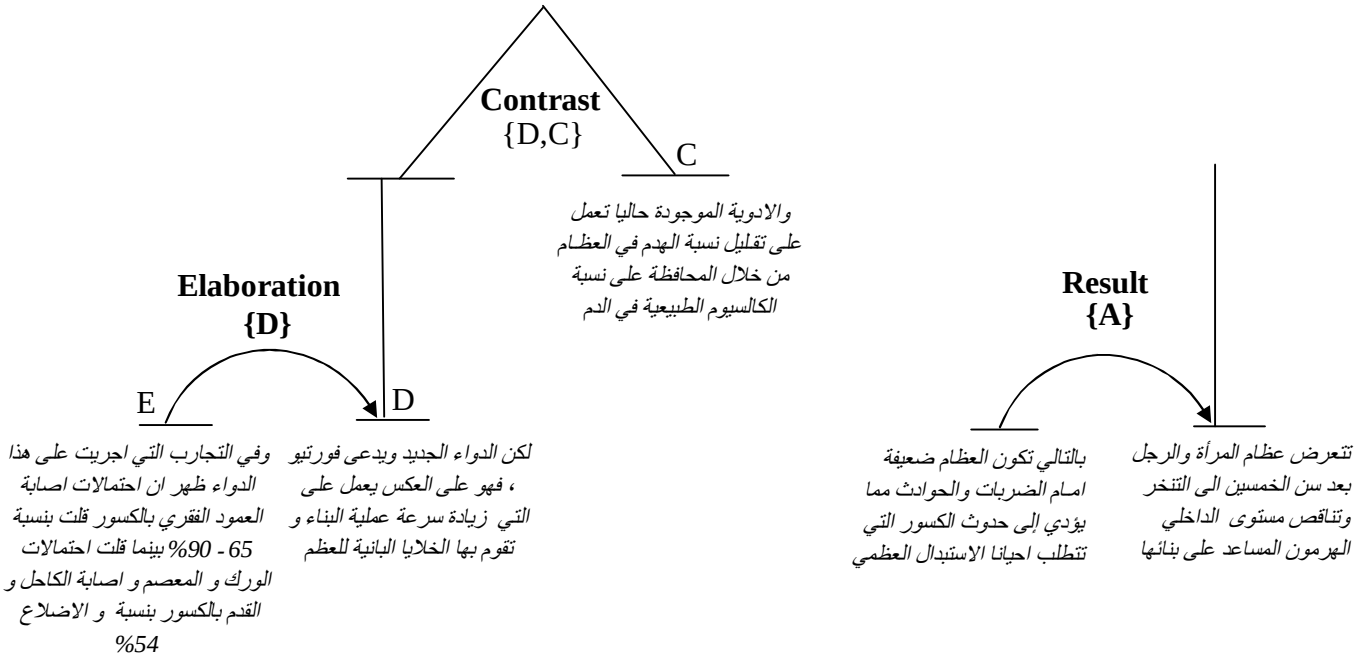

Figure 4-17: Sub-tree after applying the *Contrast* relation.

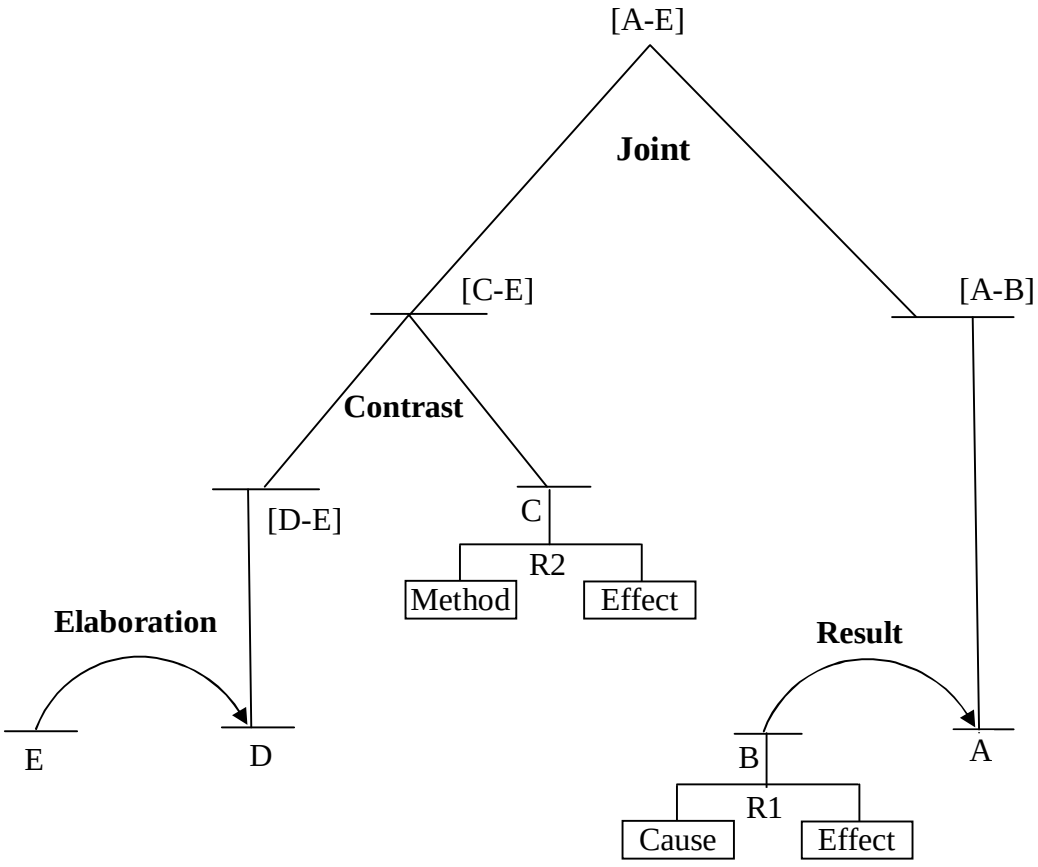

Figure 4-18: The generated Tree of text (32).

4.7 Summary

A brief explanation of RST has been presented in this chapter along with a general review of the automatic discourse parser systems that have been developed to create a full rhetorical text structure.

To address the task of Arabic text structure derivation at sentence level, the *Text Parser* model has been defined. The proposed model hypothesizes a list of adjacent and long distance rhetorical relations that may hold among sentences. The *Text Parser* model considers sentence as the basic unit of the text and incorporates the intrasentential information provided by the *Pattern Recognizer* model. Furthermore, it applies a set of heuristics to avoid any computational explosion and produces the most suitable structure representing the whole text.

The final section of this chapter has provided a worked example which illustrates how the *Pattern Recognizer* and *Text Parser* models operate together to find correct answers to "*why*" and "*how to"* questions. The next chapter presents the main component infrastructure employed by our question answering system.

Chapter 5

System Design and Implementation

5.1 Introduction

In this chapter, we provide an overview of the system infrastructure and its different components that interact to form the complete QA system. Chapter 3 and Chapter 4 have presented the two main models - *Pattern Recognizer* and *Text Parser* - that underline the task of finding answers to "*why*" and "*how to*" questions.

Arabic language differs from Indo-European counterparts syntactically, morphologically and semantically. The word representation of Arabic is rather complex because of the morphological variation and the agglutination phenomenon i.e. the large number of affixes that can be attached to a given word (Kadri and Benyamina, 1992). Accordingly, these specific characteristics should be taken into consideration when developing systems that handle Arabic texts.

Tokenization is a fundamental step in processing textual data preceding the tasks of IR, TM and several NLP disciplines. Thus, it is a pre-processing phase required to create the necessary basic knowledge, clean and structure the textual data before proceeding further with text processing. Tokenization is a language dependent approach that mainly includes normalization, stemming and stop word removal. The following three sections demonstrate different methods each of which aims at tackling a certain challenge posed when conducting an Arabic text tokenization.

5.2 Normalization

Combining characters and certain letters in Arabic texts are often spelled inconsistently which leads to multiple forms of the same word. For example, Hamzated Alif "إ ، أ" is often written without Hamza "ا". Similarly, the dotless *ya* "ى" is often confused in writing with the dotted

ya "ي"; the *ta marbota* "ة" and the *ha* "ه" are often used interchangeably when they occur in the final position of a word. This variability causes similar words such as "لان" and "لأن" to be judged incorrectly throughout the matching process. Another variation is *tatweel* which is used for decorative word elongation by expanding spaces between individual letters as in the two words "سورية" and "ســورية". The transformation of these characters into a standard form is the aim of the orthographic normalization. The following letter replacements are commonly applied by NLP researchers in order to eliminate variability.

- Convert all strings to UTF-16 encoding.
- Remove punctuation attached to words.
- Remove diacritics which represent short vowels in Arabic.
- Remove non letters.
- Replace Hamzated Alif "أ , إ" and "آ" with bare Alif "ا".
- Replace final ى with ي.
- Replace final ة with ه.

It is worth mentioning here that the *Pattern Recognizer* applies normalization only when tokens are preceded with the symbol (!) as demonstrated in Section 3.3.1.

5.3 Stemming

Stemming is a very essential technique for processing Arabic language since it is a highly inflectional one with 85% of its words derived from trilateral roots (Al-Fedaghi and Al-Anzi, 1989). Arabic roots are surrounded by a huge number of prefixes, suffixes, or both. The majority of the Arabic words (nouns, verbs, adjectives) are derived by applying a set of morphological patterns "الاوزان الصرفية" to consonantal roots to which affixes and infixes are added as shown in Table 5-1. Morphological patterns are abstractions which can be considered as an indicator of the common concept of the meaning of the word such as *tool*, an *event place/time* and *instrument* as illustrated in Table 5-2.

In this context, the root is representative of core meaning that does not account for the full meaning of a particular concept and it thus needs additional semantic features associated with a morphological pattern in order to form an Arabic word. The following terminological terms are employed in the field of linguistics to describe the representation level of a certain word:

- *Root*: The basic unit of a word that cannot be reduced into smaller constituents.
- *Stem*: the least marked form of a word. That is, it represents the uninflected word without affixes. Stems are generated by applying one of the Arabic morphological patterns on roots.
- *Lemma*: The basic dictionary-form that refers to the set of all word sharing the same meaning.

Figure 5-1 describes the process of word-formation by means of the interaction between *Root*, *Stem* and *Lemma* with derivational affixes in Arabic morphology. Table 5-3 presents some examples of the root (ك ت ب).

Stem = root + pattern
Lemma = prefix (es) + stem + suffix (es)
Word = prefix (es) + lemma + suffix (es)

Figure 5-1: Derivation levels of a certain word.

Derivation	Pattern	Root
دحرجة *rolling*	فعللة	د ح ر ج
شارب *drinker*	فاعل	ش ر ب
مكتب *office*	مفعل	ك ت ب
مكتبة *library*	مفعلة	ك ت ب
مفتاح *key*	مفعال	ف ت ح

Table 5-1: A sample of words extracted by applying morphological patterns.

Meaning		Pattern	Word
أداة	*instrument*	فِعَال	مفتاح *key*
عدة	*tool*	فَاعول	ساطور *chopper*
مكان	*event place*	فِعَل	مكتب *office*
زمان	*event time*	فِعِل	موعد *appointment*

Table 5-2: A sample of common concepts associated with morphological patterns.

Root	Stem	Lemma	Word
ك ت ب	كتاب *book*	كتابة *writing*	كتابات *writings*
ك ت ب	مكتب *office*	مكتبة *library*	مكتبتك *your library*
ك ت ب	كتاب *book*	كتاب *book*	كُتُب *books*

Table 5-3: Few derivations of the root ك ت ب.

As we have discussed above, a given Arabic word can be found in a huge number of different forms which should pose vocabulary mismatch problems between the form of a word in a query and the forms found in a textual segment relevant to that query. Consequently, researchers in the field of NLP have developed several Arabic stemmers with the aim to reduce a word to its base form. Arabic word stemming proved to be an effective technique for computational linguistic applications. The most common stemming approaches adopted by IR and QA systems are the root-based and the light stemmers. Other researchers have pointed out that N-gram stemming technique is not efficient for Arabic Text processing (Duwairi 2005; El Kourdi et al., 2004).

5.3.1 Root-based Stemming

Root-based approach attempts to find the root of a given Arabic word using morphological rules; nouns and verbs roots are derived from a few thousand of roots. A number of algorithms have been proposed for this approach (Beesley, 1996; Al-Serhan et al., 2003; Khoja and Garside, 1999). The system developed by Khoja and Garside (1999) is a leading root extraction stemmer, a comparative study for three Arabic morphological analyzers and stemmers has shown that their stemmer has achieved the highest accuracy (Sawalha and Atwell, 2008). Khoja's stemmer is an open source and makes use of several linguistic data files such as a list of diacritic characters, punctuation character and 168 stop words. Furthermore, the list of roots consists of 3800 trilateral and 900 quad literal roots.

The main drawback of root-based stemming is the over-stemming that is defined as "*taking off a true ending which results in the conflation of words of different meanings*" (Al-Shammari and Lin 2008). In other words, many words that don't have similar concept are grouped into the same root. For example, the Arabic words فراشة "*butterfly*" and يفرش "*unfold*" originate from the same root (ف ر ش) while having different semantic sense.

5.3.2 Light Stemming

Unlike the aggressive practice made by the root-based stemming, the aim of the light stemming approach is to produce the stem of a given word by eliminating a small set of suffixes and/or prefixes without dealing with infixes or recognizing morphological patterns. The most effective such stemmer has been presented by (Larkey et al., 2002) who has

introduced a group of light stemmers; she has shown performance effectiveness of a number of them that included light 1, 2, 3, 8 and 10. A comparative analysis of stemming algorithms has showed that Light 10 version has achieved the best performance (Otair, 2013). Table 5-4 shows the prefixes and suffixes lists to be removed in the Light 10 stemmer.

The main criticism to the stem-based approach is that it suffers from under-stemming representation i.e. it fails in many cases to group related word forms such as broken plural nouns and their singular forms, or past tense verbs and nouns. For example, light stemmer cannot detect the syntactic similarity between اجل "*postpone*" and تأجيل "*postponement*" since they have some affixes and internal differences.

Remove prefixes	Remove Suffixes
ال	ها
وال	ان
بال	ات
كال	ون
فال	ين
لل	يه
و	ية
	ه
	ة
	ي

Table 5-4: Strings removed by the light 10 stemmer.

In order to overcome the stemming errors and reducing stemming cost, many IR researchers raise the importance of lemma level analysis (lemmatization) emphasizing that is a very useful technique for disambiguating a word's category with minimum recourses. Lemmatization has explained by Al-Shammari and Lin (2008) as "*verbs require aggressive stemming and need to be represented by their roots. Nouns on the contrary only require light suffixes and prefixes elimination*".

Al-Shammarie and Lin (2008) have introduced a new heuristics approach to generate the Arabic lemma; she has exploited certain categories of stop words in order to identify the syntactical categories of the subsequent words, particularly nouns and verbs. The appropriate stemming level is then applied accordingly. For example, locating اسم موصول "*relative pronoun*" indicates that the following word is a verb. Table 5-5 and Table 5-6 present some of the stop words preceding verbs and nouns respectively. While the stop words preceding nouns are mainly adverbs, the ones preceding verbs have different grammatical moods. Furthermore, Al-Shammarie has employed certain syntax rules of Arabic language in recognizing word's category. For instance, if the previous word is a verb, the current word cannot be a verb since Arabic language does not permit two successive verbs to exist. Also, if a word starts with a definite article, it signals that this word is a noun.

أدوات الشرط *conditional tools*	اذا ، إن ، كلما ، لما ، من ، لو ، لولا
الأدوات الجازمة *jussive tools*	لم ، لا ، لما ، ل
الأدوات الناصبة *subjunctive tools*	لن ، كي ، أن
الأسماء الموصولة *relative pronouns*	الذي، التي ، اللذين ، ....
بعض الحروف *other particles*	سوف ، قد ، ســ

Table 5-5: Sample of stop words preceding verbs.

Stop word	English Equivalence
بعد	After
فوق	Above
امام	In front of
خارج	Outside of
قبل	Before
وراء	Behind
بين	Between
بجانب	Next to
عبر	Through
منذ	Since

Table 5-6: Sample of stop words preceding nouns.

5.4 Stop Words Removal

Stop words are words used extensively in text documents that do not contribute to the semantics of the subsequent words and have no real added value, for example, "*the*", "*and*", "*for*", "*with*" and "*by*". Thus, they are example of noise in data and they must not be included as indexing terms (Alajmi et al., 2012). In this context, neglecting stop words from

consideration can be highly important and provides a significant improvement to processing text documents due to noise reduction (Feldman and Sanger, 2007).

Stop words can be divided into two groups (Abu El-Khair, 2006); domain independent stop words lists which are created using syntactic classes regardless of the nature of the data used, and domain dependent stop words lists that can be generated using corpus statistics by calculating the total number of times in which each term appears in the documents collection. A number of studies conducted in order to define a general stop words list for Arabic language based on the structure and characteristics collected from different syntactic classes. However, there is currently no standardized list of Arabic stop words therefore researchers in the field of IR adopt their own.

Abu El-Khair (2006) has performed a comparative study of the effect of stop words elimination on Arabic IR. Three stop words lists have been experimented on an Arabic corpus created in linguistic Data Consortium in Philadelphia. The first list, *general stops list*, is based on the Arabic language structure characteristics without any additions and consists of 1377 words. The second list, *corpus based stops list*, has 359 words which have been extracted depending on words frequency. Third list, *combined stop list*, combines general and corpus-based stop list together and has resulted in 1529 words. The comparison has been conducted using different weighting schemes: TF*IDF weight, the best match weight (BM25), and the statistical language modelling (KL). Experiments have showed that the *general stop list* has performed better than the other two lists; the complete list is presented in Appendix V. The *general stop* words have been selected from the following categories (Abu El-Khair, 2006):

- Adverbs.
- Conditional Pronoun.
- Interrogative Pronouns.
- Prepositions.
- Pronouns.
- Referral Names/ Determiners.
- Relative Pronouns.
- Transforms (verbs/letters).
- Verbal Pronoun.
- Others.

It is important to emphasize that although stop words can be dropped with no harm, yet they serve syntactic functions in constructing linguistic patterns as shown in Chapter 3 and they contribute to identify syntactical category of the subsequent word as illustrated in Section 5.3.2. Accordingly, they should only be filtered out while conducting the question matching phase.

5.5 Finding the Candidate Answers

In Section 4.2.2 we have discussed how rhetorical relations and the linguistic patterns constructed in this study can be employed to find answers to "*why*" and "*how to*" questions. In practice, a list of textual units which are related by *Causal* or *Explanatory* relations is created. The textual units represent the *effect* slots of the relations discovered by the linguistic patterns or the *nucleus* parts of the relations extracted using RST.

Having all textual units along with the posed question been tokenized, the matching process is then performed between the question and all units in the list to find the most relevant textual unit so that the corresponding part of this unit is returned as a candidate answer. For example, in case of locating answers for a "*why"* question, we create a list comprising the *effect* slots of all *Causal* relations found in the relevant text. The question is matched against the list members to find the most similar slot; the corresponding part to that slot i.e. the *cause* slot is then returned as a candidate answer. We compute the similarity between the question and the list of textual units by applying the Vector Space Model and rank them in descending order.

All textual units and the posed question are represented as vectors of keywords, and the cosine similarity is measured by computing the angle between the vector representing the question and each of the vectors representing the textual units as shown in formula (5-1).

$$\text{Sim}(Q, U_i) = \text{Cosine}\ \theta_{Ui} \qquad (5\text{-}1)$$

Where:

- $\text{Sim}(Q, U_i)$: Similarity between the question and a textual unit.
- θ_{Ui}: the angle between vectors representing the question and a textual unit.

The keywords of the units are associated with weights representing the importance of the keywords in the document; likewise, the keywords of the question. The weight of a term (keyword) in a vector can be determined according to formula (5-2) (Jones, 1972).

$$Wi = tf_i * log \left(\frac{U}{uf_i} \right) \quad (5\text{-}2)$$

Where:

- ti: term *t* in textual unit *Ui*.
- tfi: frequency of term *ti* i.e. how often *ti* occurs in the textual unit *Ui*.
- uf_i: the document frequency i.e. how often *ti* occurs in the whole textual units in the candidate list.
- U: the total number of textual units in the candidate list.

The angle between two vectors is measured using formulas (5-3)-(5-6).

$$\text{Cosine } \theta_{Ui} = \frac{\mathbf{Q \bullet Ui}}{|\mathbf{Q}| * |\mathbf{Ui}|} \quad (5\text{-}3)$$

$$|Q| = \sqrt{\sum_j \mathbf{w_{Q,j}^2}} \quad (5\text{-}4)$$

$$|U_i| = \sqrt{\sum_j \mathbf{w_{i,j}^2}} \quad (5\text{-}5)$$

$$Q \bullet U_i = \sum_j \mathbf{W_{Q,j}\, W_{i,j}} \quad (5\text{-}6)$$

Thus, formula (5-7) computes the similarity comparison pair wise question and textual units, where *WQ,j* , *Wi,j* are the weights of the *jth* keyword of the question *Q* and textual unit *Ui* respectively.

$$\text{Sim } (Q, U_i) = \text{ Cosine } \theta_{Ui} = \frac{\sum_j \mathbf{W_{Q,j}\ W_{i,j}}}{\sqrt{\sum_j \mathbf{W_{Q,j}^2}\ \ \sum_j \mathbf{W_{i,j}^2}}} \quad (5\text{-}7)$$

ALGORITHM 5-1 describes the process of extracting candidate answers from a text. It takes as input one question and a sequence of textual units along with a set of relations associated with these units, and returns a set of ranked answers. The algorithm identifies the question type by the initial words in the question; these words can be any of the following: in case of "*why*" questions (لماذا ، ما هو السبب ، ما سبب ، ما الذي أدى الى ، ماالذي سبب) or in case of "*how to*" questions (كيف ، ماهي الطريقة ، ما هي الوسيلة). It then matches the question against the textual

units associated with the appropriate relations. The corresponding parts of the most relevant textual units are returned in a ranked list.

ALGORITHM 5-1: Extracting answers for a given question.

```
Input:  A question Q.
        A sequence U[n] of textual units and a list of
        relation RR that holds within and among the textual
        units in U.
Output: A set A of candidate answers.
1.      A := null;
2.      Identify the type of Q;
3.      Identify a set of relations rr in RR corresponding to
        the Q type;
4.      Match Q against the textual units U[n];
5.      For each match U_i
6.        If (U_i has a relation rr_i of one of the types in rr)
7.             sp := related part of rr_i;
8.             A := A ∪ sp;
9.          Else
10.            discard the current U_i;
11.         end If
12.       end For
13.      Sort A in descending order;
```

5.6 System Design

This section gives a brief description of the general Class diagram of our system as shown in Figure 5-2. The main class is "*QuestionAnswering*" that distributes functions over three packages. The main class in the first package is "*PatternRecognizer*" which uses the set of linguistic patterns constructed in this study to find the intrasentential relations in text, "*POS tagger*" and "*Tokenizer*" are initialized in this package in order to recognize the defined patterns. The second package has the main class "*TextParser*" that analyzes the tagged sentences obtained from the previous package and employs "*DiscourseMarkers*" to discover the intersentential relations. In the third package, class "*AnswerFinder*" initializes "*GettingKeywords*" class which in turn calls "*Stemmer*" and "*Tokenizer*" classes to get vectors of keywords that enable "*Similarity"* class to find the most relevant answers. Figure **5-3** describes the Sequence diagram.

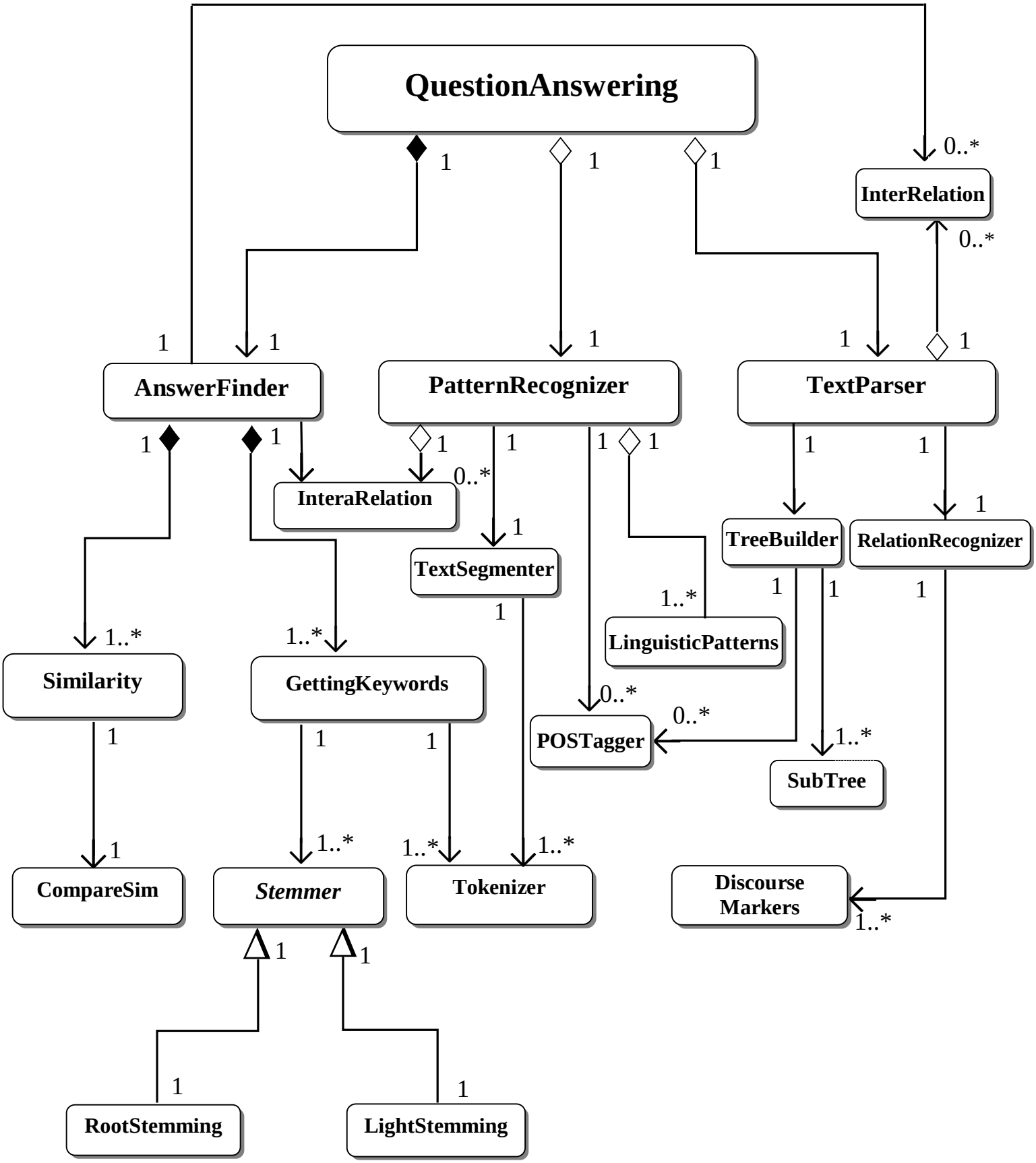

Figure 5-2: General Class diagram of the Question Answering System.

Figure 5-3: Sequence diagram of the Question Answering System.

5.7 System Implementation

System interfaces have been implemented using the JAVA programming language. The five figures below illustrate samples of the interfaces produced when asking a question related to text (32).

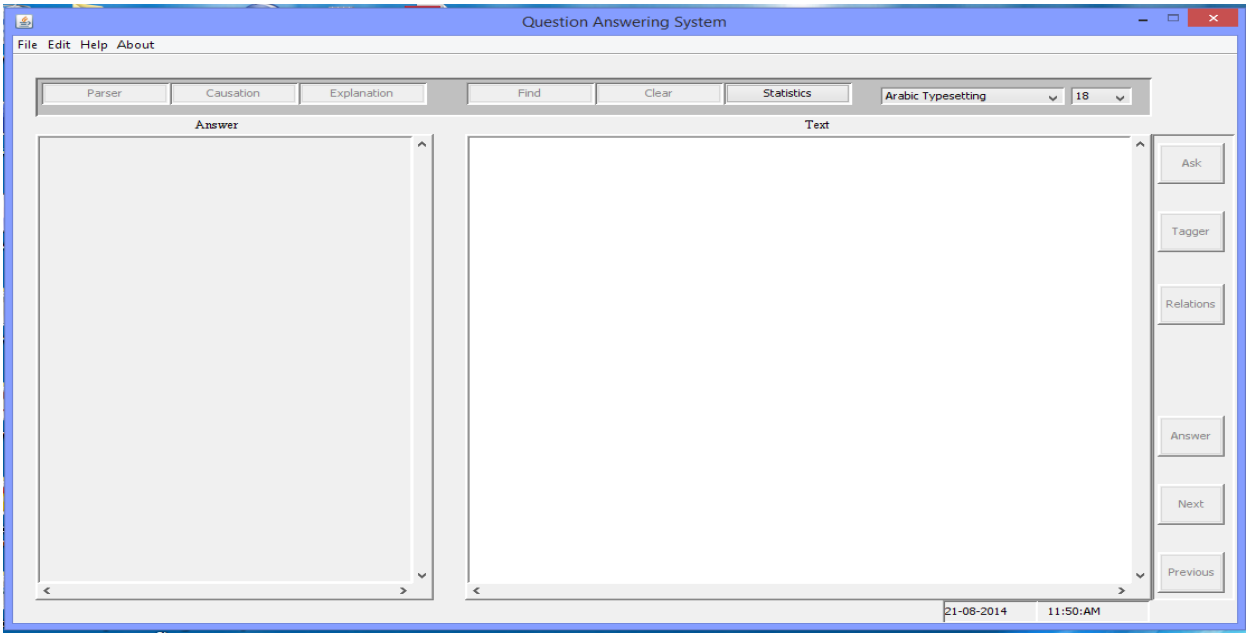

Figure 5-4: The main interface of the QA system.

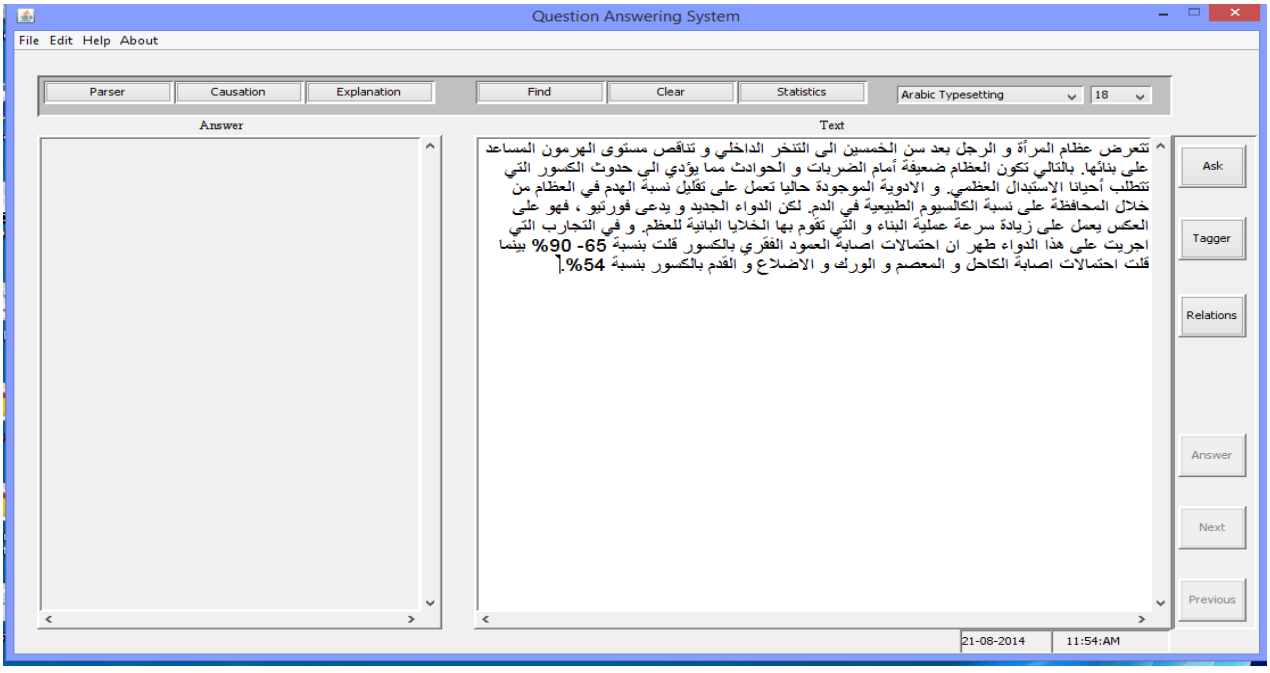

Figure 5-5: A screenshot of the system provided with text (32).

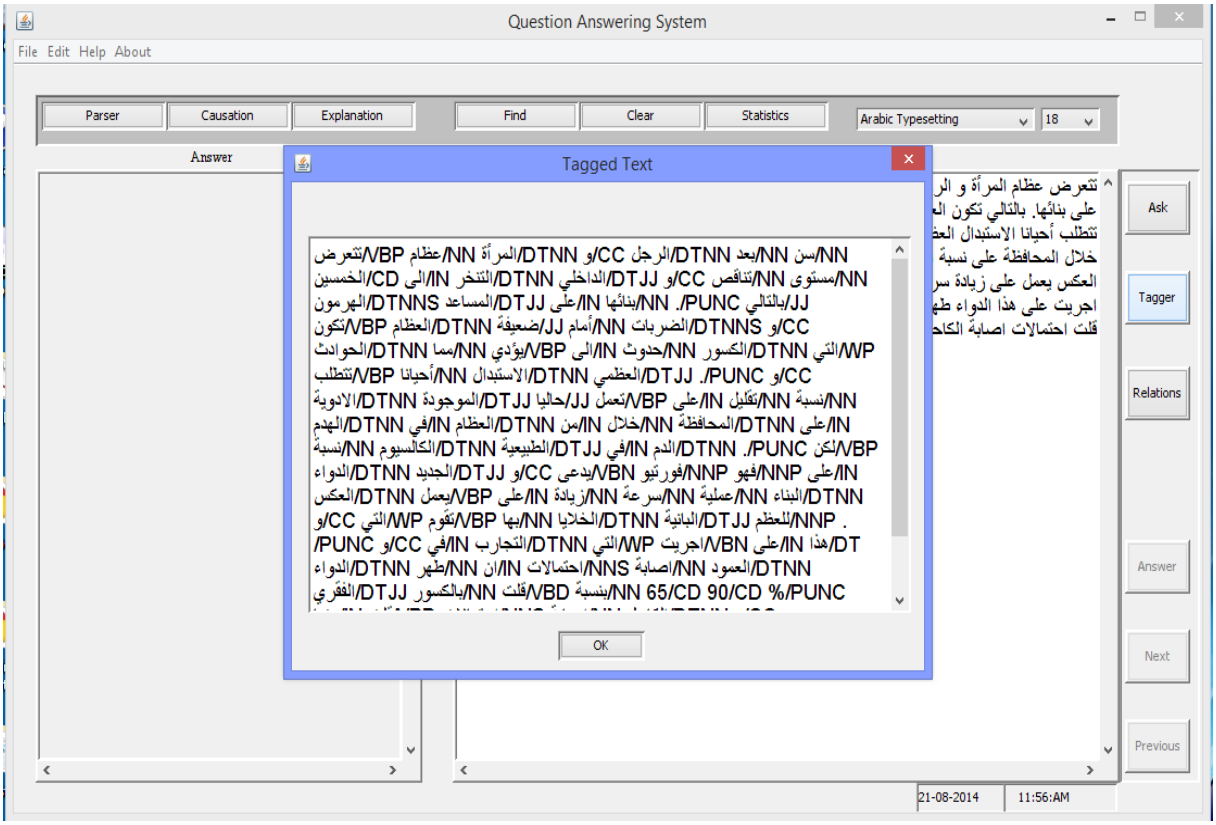

Figure 5-6: A screenshot shows the input text attached with POS tags.

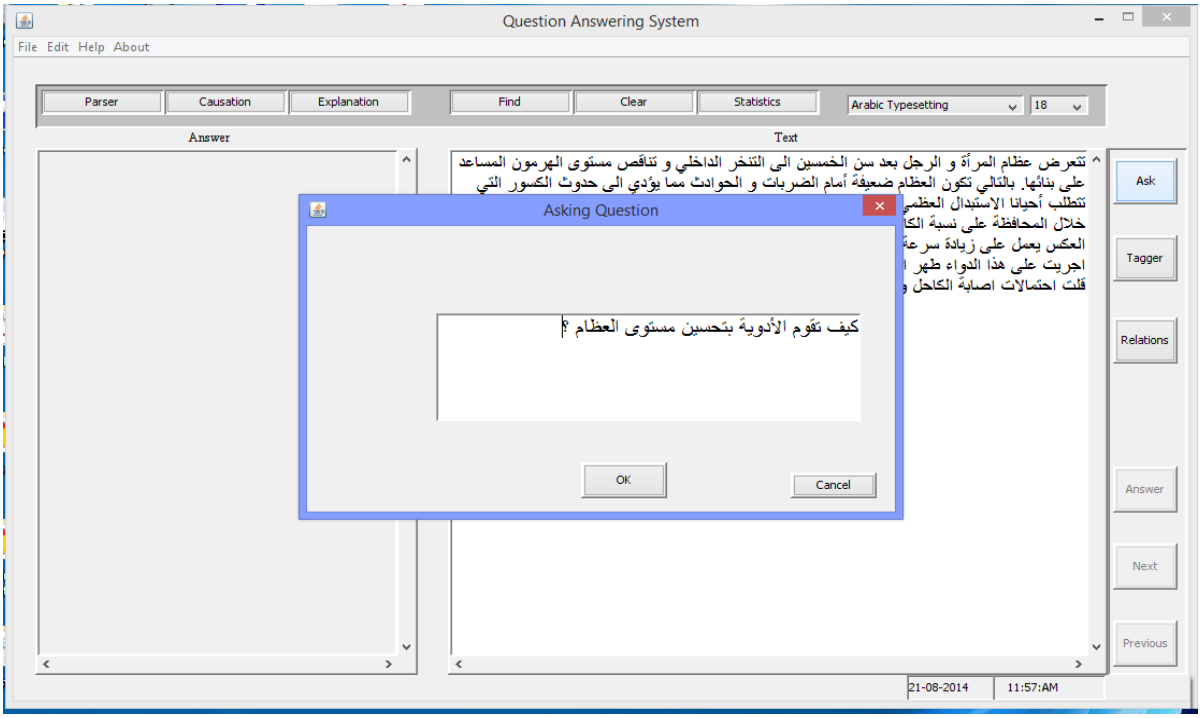

Figure 5-7: A screenshot of the form that allows users to enter a question.

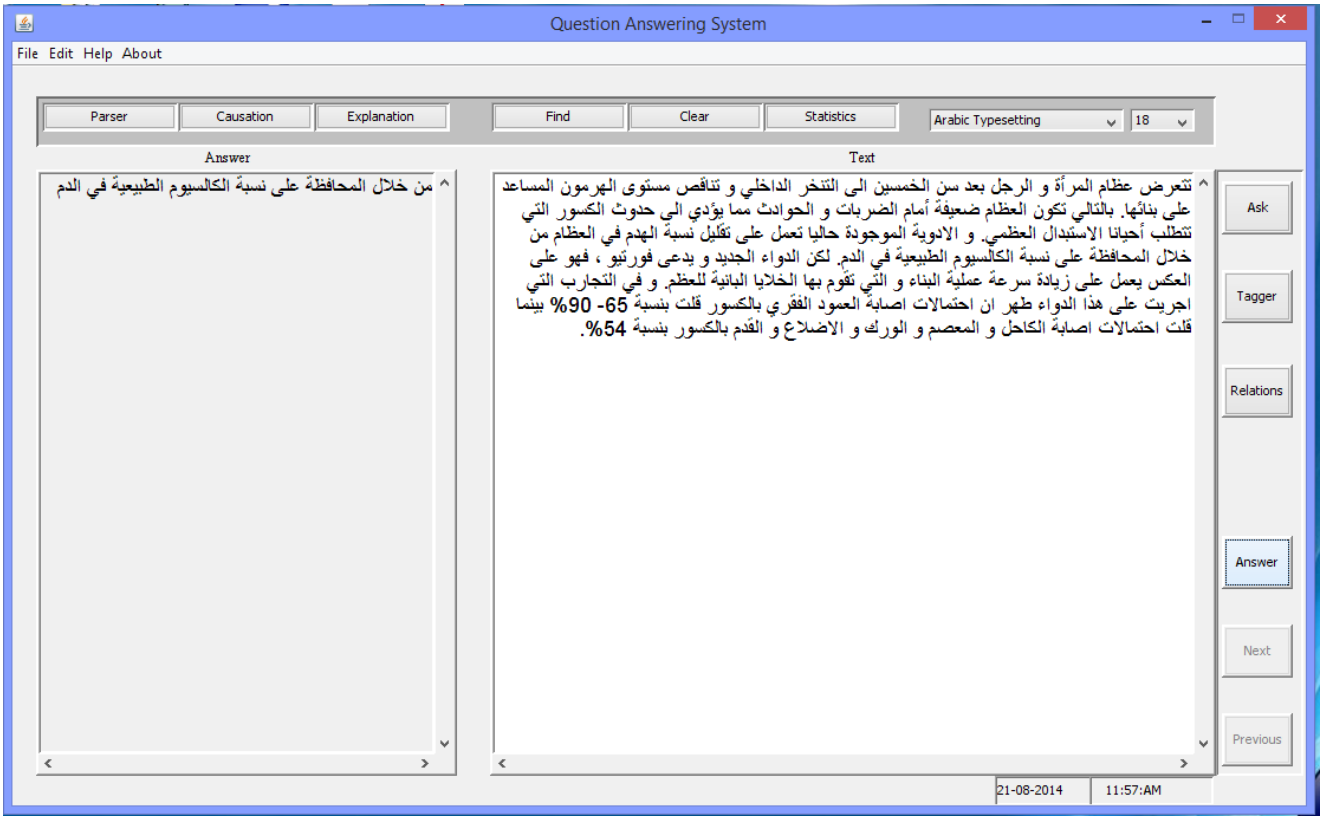

Figure 5-8: A screenshot of the returned answer.

5.8 Summary

In this chapter, we have provided our system components infrastructure. Each component represents an Arabic NLP tool with a different responsibility. This set of interacting software components have been designed to ensure that the question answering system will satisfy the Arabic language characteristics.

As a result of Arabic being a highly inflected language, stemming is a crucial technique for disambiguating word category. We have reviewed several stemmers proposed by the Arabic NLP researchers and it appeared that lemmatization has the most positive impact in this field.

This chapter has also provided the Sequence and the General Class diagrams that show how the different models and components are put together to develop our QA system. The next chapter presents the evaluation methodology and all experimental results obtained; it then revisits the research questions and summarizes the scope of this work.

Chapter 6

Evaluations and Conclusion

6.1 Introduction

The evaluations described in this chapter are divided into two parts: *Part 1* focuses on how well the linguistic patterns constructed in this study can identify the presence of intrasentential relations and the direction of these relations. Whereas, *part 2* aims at evaluating the overall performance of the question answering system. Regarding the experiment conducted in *part 2*, we follow the same strategy conducted by Verberne et al. (2007) to evaluate the appropriateness of the textual units selected by our system as candidate answers to "*why*" and "*how to*" questions.

All experiments conducted in this chapter are based on a set of articles taken from the contemporary Arabic corpus[7]. This corpus includes 415 texts written in the Modern Standard Arabic language and covers a wide range of text type. Texts are derived mainly from online magazines that publish materials produced by professional authors from different countries in the Arab world. The corpus is a useful recourse as it is readily accessible to the public and freely downloadable.

We collected the articles specifically from the categories of Health and Science & Technology of 485-2138 words each. Five independent subjects whose first language is Arabic were involved in the experiments. All the subjects are highly educated; three of them are studying languages on a doctorate level while the other two are specialists in the field of communication. In both parts of the experiments, the evaluation was performed by comparing the output generated by the system against the judgments of the subjects.

[7] http://www.comp.leeds.ac.uk/eric/latifa/research.htm

6.2 Evaluation of the Linguistic Patterns

This part of the experiments was carried out on two stages. In the first one, only the linguistic patterns were employed for discovering intrasentential relations, while in the second stage justification particles were also incorporated. As discussed in Section 3.4, justification particles- *Purpose Lam*, *causation faa* and *causation baa* - are highly ambiguous; therefore, we wish to see how they affect the performance by comparing the results obtained including them and results obtained without using them.

Eleven texts were manually segmented based on the occurrence of the full stop and this is resulted in a total of 415 sentences. Three participants were asked to read the sentences and identify the presence of *Causal* relations that are explicitly expressed in each single sentence together with the fractions representing *cause* slots and fractions representing *effect* slots. This resulted in collecting a total of 240 *Causal* relations. The *Pattern Recognizer* was then applied to extract the same information.

The performance measures used are *recall* and *precision*. Recall, in this context, is the proportion of the relations identified by the subjects that are also identified by the *Pattern Recognizer*. Precision is the proportion of relations identified by the *Pattern Recognizer* that are also identified by the subjects. Table 6-1 and Table 6-2 show the number of relations identified by the subjects for each text of the Health texts excluding and embedding the justification particles algorithms respectively; the second column presents the number of relations discovered by the *Pattern Recognizer* correctly. Table 6-3 and Table 6-4 show the same information for the Science & Technology texts. The *Pattern Recognizer* obtained a maximum overall *recall* of 78% for the Heath texts and 84% for the Science & Technology texts.

Table 6-5 and Table 6-6 display *recall*, *precision* and the corresponding F-scores for the texts belonging to the Health category excluding and embedding the justification particles algorithms respectively. Table 6-7 and Table 6-8 present the same measures for the texts belonging to the Science & Technology category. F-scores were computed using the formula (6-1). The F-score is always a number between the values of *recall* and *precision*.

$$F = \frac{2 \text{ X } \boldsymbol{Precision} \text{ X } \boldsymbol{Recall}}{\boldsymbol{Precision + Recall}} \qquad (6\text{-}1)$$

	Manually	Matched	Recall
Text 1	19	12	0.63
Text 2	32	14	0.44
Text 3	33	10	0.30
Text 4	18	11	0.61
Text 5	11	4	0.36
Overall	113	51	0.45

Table 6-1**:** Number of relations identified in the Health texts excluding the justification particles algorithms.

	Manually	Matched	Recall
Text 1	19	16	0.84
Text 2	32	28	0.87
Text 3	33	23	0.70
Text 4	18	13	0.72
Text 5	11	8	0.73
Overall	113	88	0.78

Table 6-2**:** Number of relations identified in the Health texts including the justification particles algorithms.

	Manually	Matched	Recall
Text 1	18	5	0.27
Text 2	18	12	0.66
Text 3	27	15	0.55
Text 4	7	5	0.71
Text 5	24	13	0.54
Text 6	33	24	0.73
Overall	127	74	0.58

Table 6-3**:** Number of relations identified in the Science & Technology texts excluding the justification particles algorithms.

	Manually	Matched	Recall
Text 1	18	16	0.89
Text 2	18	15	0.83
Text 3	27	24	0.88
Text 4	7	6	0.86
Text 5	24	17	0.71
Text 6	33	29	0.88
Overall	127	107	0.84

Table 6-4**:** Number of relations identified in the Science & Technology texts including the justification particles algorithms.

	Recall	Precision	F – Score
Text 1	0.63	0.95	0.76
Text 2	0.44	0.97	0.61
Text 3	0.30	0.91	0.45
Text 4	0.61	0.98	0.75
Text 5	0.36	0.94	0.52
Overall	0.45	0.95	0.61

Table 6-5**:** Precision, Recall and F-score for the Health texts excluding the justification particles algorithms.

	Recall	Precision	F-Score
Text 1	0.84	0.86	0.85
Text 2	0.87	0.84	0.85
Text 3	0.70	0.74	0.72
Text 4	0.72	0.88	0.79
Text 5	0.73	0.67	0.70
Overall	0.78	0.80	0.79

Table 6-6**:** Precision, Recall and F-measure for the Health texts including the justification particles algorithms**.**

	Recall	Precision	F-Score
Text 1	0.27	0.93	0.42
Text 2	0.66	0.96	0.78
Text 3	0.55	0.93	0.69
Text 4	0.71	0.95	0.81
Text 5	0.54	0.94	0.69
Text 6	0.73	0.85	0.76
Overall	0.58	0.93	0.71

Table 6-7: Precision, Recall and F-measure for the Science & Technology texts excluding the justification particles algorithms.

	Recall	Precision	F – Score
Text 1	0.89	0.80	0.84
Text 2	0.83	0.75	0.79
Text 3	0.88	0.75	0.81
Text 4	0.86	0.88	0.87
Text 5	0.71	0.74	0.72
Text 6	0.88	0.71	0.79
Overall	0.84	0.76	0.80

Table 6-8: Precision, Recall and F-measure for the Science & Technology texts including the justification particles algorithms.

Figure 6-1, Figure 6-2, Figure 6-3 and Figure 6-4 illustrate how excluding and embedding the justification particles as indicator of *Causal* relations impact *recall*, *precision* and F-score. We observe that incorporating the justification particles algorithms boosts the efficiency of the *Pattern Recognizer* by a large margin improving the overall *recall* by 33% for Health texts and 26% for Science & Technology texts. However, employing the justification particles algorithms gave the rise to the number of instances where the *Pattern Recognizer* mistakenly indicated the presence of *Causal* relations. Accordingly, the overall *precision* degraded by

15% for Health texts and 17% for Sciences and Technology texts. The main reason for that is unsurprisingly the number of errors accounted by the justification particles algorithms. These particles are ambiguous tools and can play different roles other than causation indicators.

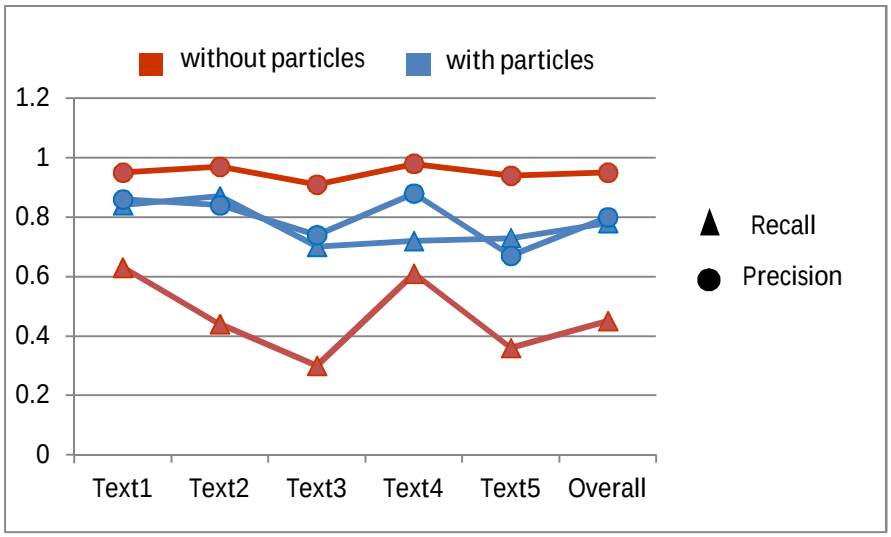

Figure 6-1: Recall and Precision for the Health texts.

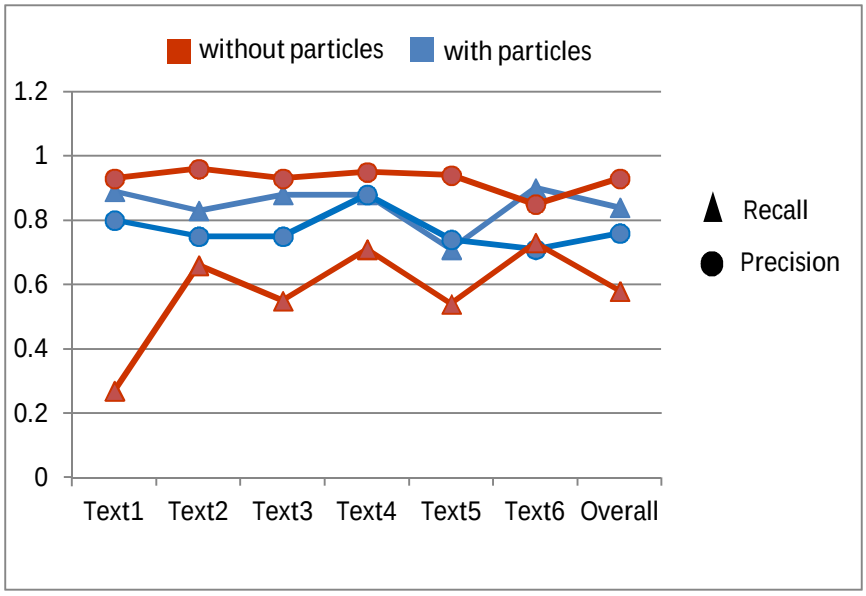

Figure 6-2: Recall and Precision for the Science & Technology texts.

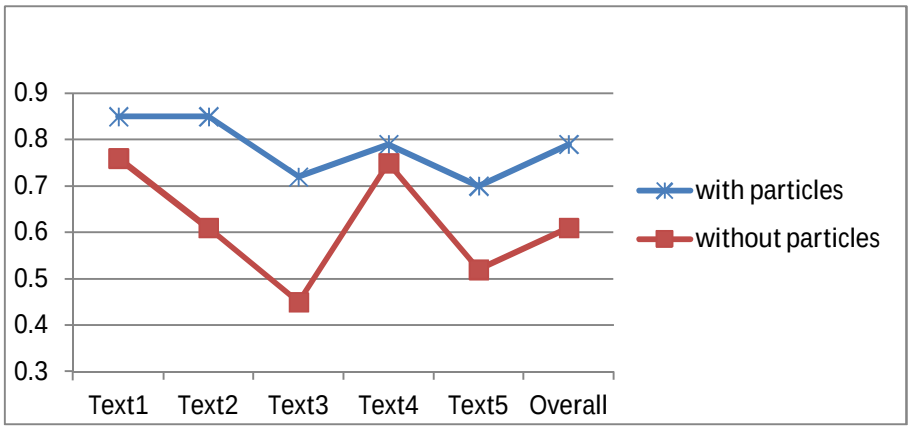

Figure 6-3: F-Score for the Health texts.

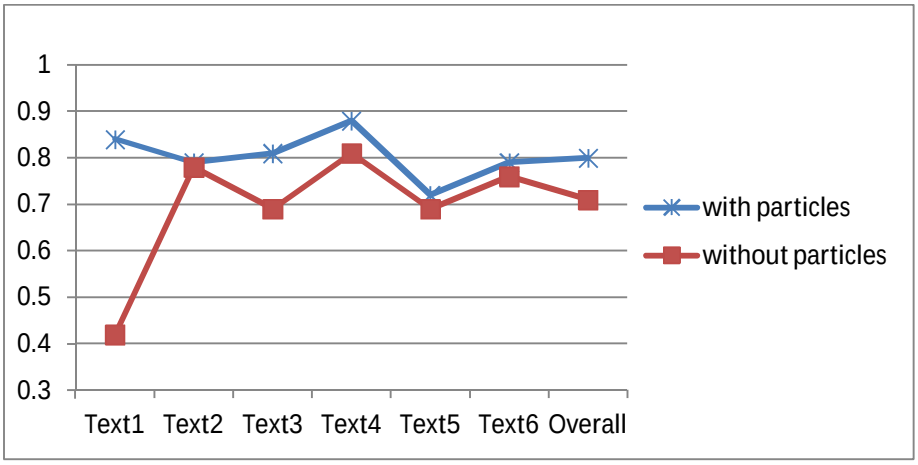

Figure 6-4: F-Scores for the Science & Technology texts.

On examining the relations set which the *Pattern Recognizer* failed to identify in the second stage of the experiments i.e. incorporating linguistic patterns and justification particles algorithms, we found out that 30 relations of the set (67%) were missed because of particular kinds of linking words that were not included in the list of patterns. Some of these linking words are rarely used for indicating such relations. Texts (35) – (36) are two examples of the sentences containing relations that are not picked out by the *Pattern Recognizer*.

The other set of the relations (15 relations), which the *Pattern Recognizer* was unable to discover, was due to unexpected sentence construction. This group covers 33% of the all missed relations. The causal relation resides in sentence (37) is one example. Indeed, this type

of relation is indicated implicitly and inferred from the world knowledge needed to identify such relations.

(35) [وبعد اجراء تحقيقات]^{M} [اكتشف أن الشركة المصدرة صينية.]^{E}

"[Investigations]^{M} [revealed that the export company is Chinese]^{E}"

(36) ...ولكن السر الذي يخفى على جميع السيدات وخاصة ذوات البشرة الزيتونية أو العربية هي أن [جميع هذه المستحضرات والعلاجات ذات فائدة محدودة]^{E} ما لم ندرك [دور أشعة الشمس الخليجية في إفشال الكثير من هذه العلاجات.]^{C}

"[But the secret, which is hidden from most women especially those with olive complexion or Arab complexion, is that all of these products and treatments are of short-term effects]^{C} [unless we become aware that the Gulf sunlight spoils most of such treatments.]^{E}

(37) [وهناك ايضا العلاجات البديلة التي توفر احيانا راحة للمريض من العلاجات الكيماوية،]^{C} [وبعض هذه العلاجات اصبح مقبولا من قبل الوسط الطبي.]^{E}

"[There are also the alternative therapies which, sometimes, work as a good substitute of chemical therapies,]^{C} [and some of these therapies have become acceptable in the medical field]^{E}"

6.3 Evaluation of the QA System

For the purpose of evaluating the performance of the QA system, we distributed Arabic articles to five subjects and asked them to read some of the texts and formulate "*why*" and "*how to*" questions for the answers that could be found in the text, the subjects were also asked to formulate answers to each of their questions. This resulted in a total of 90 question-answer pairs (70 *why* questions and 20 *how to* questions).

We ran our system on the 90 questions we collected, and then compared the answers found by the system to the user-formulated answers; if the answer found matches the answer formulated by the subject then we consider the answer found as correct. The system was able to return the correct answer for 61 questions and this means that the system obtained a *recall* of 68%. An overview of the system results is given in Table 6-9.

	# questions	% of all questions
Questions handled	90	100
Correctly answered	61	68
Wrongly answered	29	32

Table 6-9: The outcome of the QA system.

As for the questions that the system couldn't extract correct answers out of them, they are placed in two categories of questions. First, questions where there are no explicit relations between the textual units representing the question and the textual units of the answers. This category comprises 5 questions (18% of the questions had not been answered correctly). Questions in this category are connected to the answers spans with relations expressed implicitly in text. For example, question (38) posed by one of the subjects refers to sentence (39) in the source text. This question corresponds to the string " حصل على منحة بمقدار مليون دولار من قطاع صناعة القطن في كاليفورنيا" which is embedded in the subject-formulated answer "لكي يحاول تعديل فراشة القطن الزهرية وراثيا". In such a case, the system is unable to identify the location in the text where the two parts of a relation are linked. Using general knowledge, the reader has no difficulty inferring that Miller has been granted million dollars for the purpose of conducting his research.

(38) لماذا حصل ميللر على منحة قدرها مليون دولار ؟

"Why did Miller get a grant worth of one million dollars?"

(39) ويحاول ميللر، الذي حصل على منحة بمقدار مليون دولار من قطاع صناعة القطن في كاليفورنيا ، تعديل فراشة القطن الزهرية وراثيا لكي تكون نشطة جنسيا ولكن غير قادرة على التناسل بالطريقة المناسبة.

"Miller, who's got a grant worth of one million dollars from Cotton industry sector in California, endeavours to genetically modify the pink cotton butterfly to be sexually active but unable to reproduce in a proper way"

The other category (24 questions, 82% of the all failed questions), are the cases where the linguistic items indicating the relations were not supported by the *Pattern Recognizer* or the *Text Parser*. Consider for example, one of the failed questions (40) which refers to sentence (41). In this sentence, the word "تخفيفا" "to alleviate" which belongs to the syntactic category مفعول لأجله "Accusatives of purpose" signals the presence of a *Causal* relation. Generating answers based on the occurrence of a specific POS indicator requires full syntactical parsing. Certainly, the set of missed intrasentential relations which discussed in Section 6.2 impacts the performance of the QA system. For example, the following question " لماذا تتسم العمليات التجميلية بمحدودية الفائدة" was not answered correctly due to its correlation with the causal relation contained in sentence (36). This relation was not discovered by the *Pattern Recognizer*.

(40) لماذا يرتدي المريض نظارات شمسية بعد وضع قطرة الاتروبين

"Why do patient wear sunglasses after using Atropine eye drops?"

(41) تقوم قطرات أو مراهم الاتروبين بتوسيع حدقة عين المريض وسوف يلاحظ المريض بعد وضع القطرة أن الأشياء القريبة من يده تصبح غير واضحة كما انه سينزعج من ضوء الشمس ، وقد يكون من الضروري وضع نظارات شمسية تخفيفا لشعوره بعدم الارتياح.

"Atropine drops or creams help make the pupil of the eye larger and after using the drops, the patient will notice that close things become blurred; and the sunlight will be a source of annoyance. Therefore wearing sunglasses might be essential **to alleviate** this unpleasant sensation."

Figure 6-5 illustrates the distribution of the questions answered correctly (green coloured partitions) together with the failed questions (red coloured partitions). Nearly 55% of the questions were answered correctly based on the indication of intrasentential relations, whereas correct answers for 13% of the questions correlate to the presence of rhetorical relations between sentences.

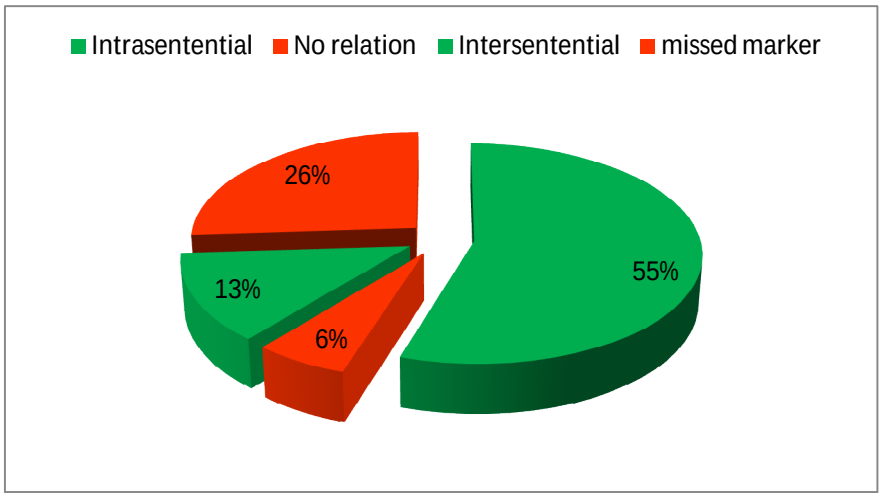

Figure 6-5: The distribution of the questions test.

6.4 Conclusion

The main motivation behind the work in this thesis was to consider simple techniques with the aim of finding answers for "*why*" and "*how to*" questions where both could be easily understood and operate quickly. We envisage that this work would fill a gap in the field of Arabic QA Systems. To this end we introduced the two analytical models: *Pattern Recognizer* and *Text Parser* which were built to be performed with high accuracy and low complexity. We summarize them in this section, adding emphasis on the evaluation results.

6.4.1 Identifying the Intrasentential Relations

In Section 2.5.1, we investigated different studies which tackled mining causation in texts written in languages other than Arabic. A number of these studies used hand-coded pattern and specific knowledge bases. Other systems employed machine learning approaches in order to automatically construct syntactic patterns.

Researchers employing machine learning techniques made use of knowledge resources available for the language they addressed, e.g. (large annotated corpora, WordNet, Wikipedia etc.). Such resources provide externally verified analyses of POS and constituency, and are invaluable for those desiring to evaluate and train models that involve statistical component. Given a similar corpus of Arabic texts annotated with *Causal* and *Explanatory* relations, it should be possible to automatically acquire patterns. To our knowledge, the only available resource annotated with discourse relations for Arabic is the LADTB, a corpus that contains approximately 500 causal relations.

However, the morphological representation of Arabic is rather complex because of the morphological variation and the agglutination phenomenon (Kadri and Benyamina, 1992). Furthermore Arabic is a highly inflectional language with 85% of words derived from trilateral roots surrounded by a huge number of prefixes, suffixes, or both. These prefixes and suffixes could be associated with any type of Arabic word such as noun, verb, adjective etc.

In fact, it is challenging to capture the syntactical arrangement of many of the causative connectors. Consider for example the lemma "سبب" which can be represented by a variety of syntactical forms. More than 150 occurrences of this connector in different sample documents were investigated in order to generate the patterns group that can accurately identify *cause* and *effect* information associated with the set of words belong to the lemma "سبب". Table 6-10 shows samples of the generic structure for sentences involving the lemma "سبب".

Accordingly, machine learning approaches followed in research presented in Section 2.5.1 could not be applied in this study due to lack of large quantities of annotated data. Hand-crafting in order to construct linguistic patterns able to indicate semantic relations within sentences is therefore still necessary.

العوامل المسببة [E] هي [C]
وبسبب [C] [E]
[C] سببت [E]
السبب وراء [E] [C]
[E] كان سببها [C]
[C] احد الاسباب [E]
[E] والسبب [C]
السبب في [E] [C]

Table 6-10: Samples of the generic structure of sentence contain "سبب".

To our knowledge, this work represents the first attempt in the field of Arabic NLP for identifying and extracting causal and explanatory information within sentences. To reach this goal, we established the *Pattern Recognizer* model based on a set of linguistic patterns. The model was built to address the first question in this thesis: *Is it possible for hand-crafted patterns to convey information using a little of NLP techniques*?

The constructed patterns were generated by analyzing a collection of data extracted from a large untagged Arabic corpus called ***arabiCorpus***. We surveyed Arabic studies that considered the linguistic items indicating causation and explanation at sentence level. The pattern development process went through several steps of reasoning method in which the patterns cycled between the Inductive and Deductive phases until we developed a set of approximately 900 linguistic patterns. Moreover, three independent algorithms were proposed in order to discover the causal/explanatory role that may be indicated by the justification particles: ***Purpose Lam*** (لام التعليل) – ***causation faa*** (فاء السببية) and ***causation baa*** (باء السببية).

The *Pattern Recognizer* model was evaluated on eleven articles taken from Health and Science & technology domains. With the participation of human judges, a total of 240 *Causal* relations were manually identified. The linguistic patterns were then applied together with the justification particle algorithms. Under this condition, the results showed that about 81% of the relations that were clearly expressed in the 11 articles could be correctly identified and extracted. Of the instances that the *Pattern Recognizer* identified as intrasentential relations, about 78% were correct. The majority of the wrong instances were picked out by one of the justification particles algorithm as they were highly ambiguous. Ignoring these particles and

applying only the linguistic patters has improved the *precision* by 16%. However, this improvement comes at the cost of the *recall* measure which is reduced by 29% demonstrating that this type of particle plays a key role as intrasentential indicator.

Utilizing a full syntactical parser and performing word sense disambiguation, especially for justification particles, can substantially reduce the errors associated with the *precision* measure. Also, if a full syntactical parser is used, the linguistic patterns can be made much simpler and fewer patterns need to be used. This will definitely come at the cost of computational complexity.

As the *Pattern Recognizer* model obtained a maximum overall *recall* of 81% we conclude that using the linguistic patterns boosted with the justification particles algorithms will be effective for identifying intrasentential information. Furthermore, the extracted linguistic patterns reflect strong relation indicators and constitute a useful feature in the future for systems adopting machine learning techniques in acquiring patterns that signal causation and explanation.

6.4.2 Automatic Derivation of the Arabic Text Structure

Identifying discourse relations is a crucial step in discourse analysis. It is considered useful for many applications in both language and speech technology. Automatic identification of coherent relations has gained popularity in the literature within different theoretical frameworks.

In Chapter 4, we provided an overview of RST which shapes the framework of our QA system. RST has been utilized in many computational linguistic applications and has proven to be an authentic tool for analyzing the structure of coherent texts. Furthermore, human annotators show considerable consensus which implies that the rules for assigning the rhetorical relations are clearly defined (Bosma, 2005).

Section 4.3 presented some background information on previous RST systems that were dedicated to the automatic extraction of discourse structure on full scale. Most of them were oriented to the English language. The only attempt for deriving Arabic discourse structure was presented by Mathkour, Touir and Al-Sanea (2008) where they identified eleven rhetorical relations. They adopted Marcu's (2000a) methodology and adapted it to be used in

developing an Arabic text summarizer. However, the use of their discourse parser was restricted to small texts of around (30-35) lines due to the high computational complexity involved in processing a large number of hypothesized relations associated with large texts.

In this study, we built our discourse parser on top of the output obtained from the *Pattern Recognizer* which is sentences that are already annotated with intrasentential relationships. To fulfil this goal, we developed the *Text Parser* model that would approach text from a discourse perspective. The *Text Parser* is meant to break away from the sentence limit imposed on the *Pattern Recognizer* and emphasize the strategies employed above these limits to hold the whole text together as a unit. Furthermore, the *Text Parser* is led by a set of heuristic scores to avoid any computational explosion.

In this regard, DMs play a key role in indicating discourse relations between sentences. They link segments of discourse together to achieve coherence and cohesion. The connective functions of DMs have been heavily emphasized by a number of linguists (Schiffrin, 1987; Blakemore, 1996; Fraser, 1999). The main problem in studying DMs in any natural language is that they have multiple functions. Here, there are conflicting views in approaching these items making it impractical to adopt an exhaustive list. Therefore, each researcher has to choose his own DMs list in consonance with the objective of his study.

The *Text Parser* employs a sub list of the DMs proposed by Al-Kohlani (2010) who investigates DMs in Arabic newspaper opinion articles. Her DMs are particularly useful for our research since she employed RST in generating them. DMs described in Al-Kohlani's (2010) work are bidimensional i.e. they operate at more than one level of discourse structure. Out of her main list we chose those associated with sentence level.

It is of interest to find out how effective is the use of the *Text Parser* and the *Pattern Recogniser* models in the task of extracting answers to "*why*" and "*how to*" questions; and this would answer the second question in this thesis: *To what extent can discourse analysis help in selecting answers to "why" and "how to" questions for the Arabic language*. We asked five subjects to read Arabic texts and formulate "*why*" and "*how to*" questions for the answers that could be found in the text. The subjects were also asked to formulate answers to each of their questions. This resulted in a total of 90 question-answer pairs.

For 68% of the questions, the system was able to find the correct answer. In a large majority of the failed cases (82%) the system was unsuccessful due to the misidentification of the linguistic items indicating these answers. These items include DMs that missed by the *Pattern Recognizer* and types of POS labels that require a syntactical parser to be incorporated. For the other cases of the questions that couldn't be answered (18%), the system was not able to find an explicit relation between the textual units representing the question and the textual units of the answers.

The main drawback to the test data collection method, as stated by Verberne et al. (2007), is that the questions were gathered from subjects who have been reading a text. This involves the risk that the subjects might have been tempted to invent "*why*" and "*how to*" questions which has led to a set of questions that is not completely representative of a user's real information need. Another limiting factor is that the subjects tend to use the same terms as those occurring in the texts. Such an overlap may not be possible in the natural questions.

It is hoped that the new frameworks proposed in this thesis will advance the field of TM for the Arabic language, giving rise to the Arabic systems that answer "*why*" and "*how to*" questions which can be used by the general public to access the growing source of knowledge available as free text.

6.5 Future Directions

Whilst the approaches to QA introduced in this thesis help answering two main questions i.e. *why* and *how to* questions, they could be improved to cover more points of research related to this field. This section briefly considers a number of directions where further research can depend on to enhance these approaches to be applied on non factoid questions.

6.5.1 Intrasentential Relations

This study made use of discourse connectives as indicators of the presence of *Causal* relations. Causal relation can also be expressed using some types of verbs. Such types are called *causative verbs* which their meaning implicitly induce causal elements. For example, the two transitive verbs يولد "Generate" and يقتل "Kill" in sentences (42) and (43) can be paraphrased using the intransitive words "*die*" and "*happen*" respectively as "*to cause to die*" and "*to cause to happen*". Writers have different views on how to distinguish *causal verbs*

from other transitive verbs that are not causal. Gonsalves (1986) pointed out that the *causal verb* indicates that the agent (i.e. the subject of the verb) participates crucially in the causation by his acts. As an extension to this work, we plan to explore the use of *causal verbs* in the Arabic literature.

(42) قال محللون ماليون ان تحرك الاسهم في نطاق تذبذب ضيق خلال الاسبوع الماضي ولد حالة من الرتابة والملل في سوق الاستثمار.

"Financial analysts believe that the fluctuation of the stock market movement within a narrow range during last week **generated** a state of monotony in the investment market"

(43) العسل دواء طبيعي يقتل البكتريا دون أن يصيب الأنسجة بالضرر.

"Honey is a natural medicine that **kills** bacteria without damaging the tissues."

6.5.2 Text Structure Derivation

In the course of this study, all texts structures were generated in the framework of RST. Therefore, the *Text Parser* model was forced to build tree-like representations that subsumed all the discourse units in a text. In contrast, Wolf and Gibson (2005) took a less constrained approach stating that *"trees are not a descriptively adequate data structure for representing discourse structure"*. They allow annotators to make explicit coherent relations that hold between any two textual units in a text. For example, text (44) divided into discourse segments was presented by Wolf and Gibson (2003) to justify their approach. By applying this protocol, text structures look like graphs more than trees. This can be illustrated by the *Elaboration* relation between segments [4-5] and segment [2] which crosses the *Attribution* relation between segment [3] and segments [1-2] as shown in Figure 6-6.

Wolf and Gibson used their analysis as foundation for psycholinguistic research as well as information extraction. A future study might investigate if utilizing such framework would show improvements in recognizing distance relations.

(44) [Farm prices in October edged up 0.7% from September][1] [as raw milk prices continued their rise,][2] [the Agriculture Department said][3] [Milk sold to the nations' dairy plants and dealers averaged $14.50 for each hundred pounds,][4] [up 50 cents from September and up $1.50 from October 1988,][5] [the department said.][6]

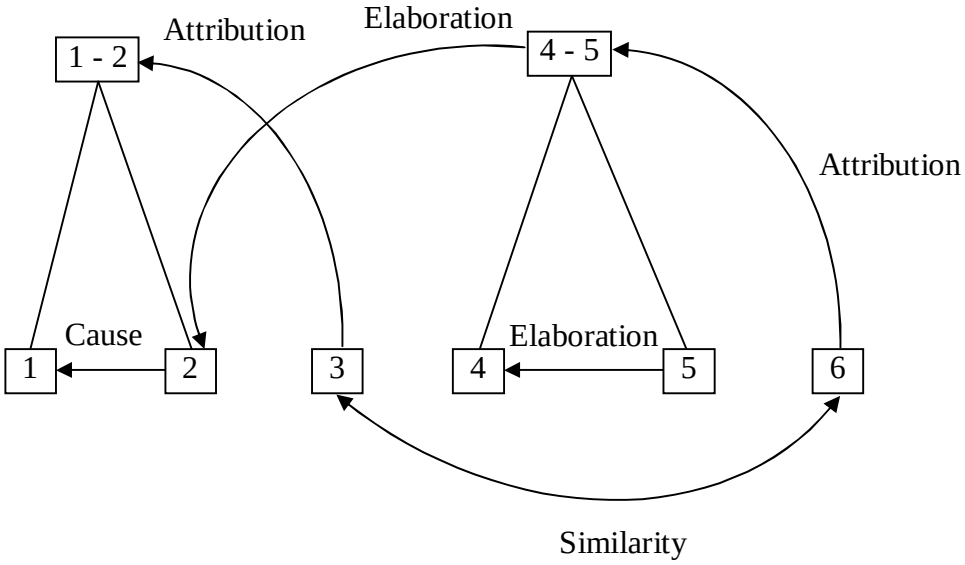

Figure 6-6: A graph representation of text (44).

6.5.3 System Evaluation

As we pointed out in Section 6.3, the test data were collected through elicitation which implies that questions might have been influenced by the same linguistic cues used by the text producers. This results in lexical overlap more than one would expect for natural questions. It is important to remember, however, that the ultimate goal of question answering systems is to find answers in vast amounts of information which users might not have access to. Future work should be dealing with questions formulated independently of a specific text. To reach this goal consideration must be given to the query expansion techniques such as those discussed in Section 2.3.1.

Appendix I
List of Abbreviations

Answer Processing	AP
Bag Of Words	BOW
Discourse Marker	DM
Elementary Discourse Unit	EDU
Fact Based Question Answering	FBQA
Information Extraction	IE
Information Retrieval	IR
Maximum Entropy	ME
Mean Reciprocal Rank	MRR
Modern Standard Arabic	MSA
Named Entity	NE
Named Entity Recognition	NER
Natural Language Processing	NLP
Non-Factoid Question Answering	NFQA
Part Of Speech	POS
Passage Retrieval	PR
Penn Discourse TreeBank	PDTB
Question Answering	QA
Reciprocal Rank	RR
Relational Database Management System	RDBMS
Rhetorical Structure Theory	RST
RST Discourse Treebank	RST-DT
Segmented Discourse Representation Theory	SDRT
Text Mining	TM
Text REtrieval Conference	TREC

Appendix II

Regular Expressions Symbols/Operators

\s	Any whitespace
\w	Any word character
\d	Any digit
\b	A word boundary
^	Anything except occurrences of the pattern
*	Matches zero or more occurrences of the pattern
+	Matches one or more occurrences of the pattern
?	Matches zero or one occurrences of the pattern
{n}	Matches exactly n occurrences
{n, m}	Matches between n and m (inclusive) occurrences

Appendix III

UTF-16 Encoding for Arabic Letters

0621	ء
0622	آ
0623	أ
0624	ؤ
0625	إ
0626	ئ
0627	ا
0628	ب
0629	ة
062A	ت
062B	ث
062C	ج
062D	ح
062E	خ
062F	د
0630	ذ
0631	ر
0632	ز
0633	س
0634	ش
0635	ص
0636	ض
0637	ط
0638	ظ
0639	ع
063A	غ
0641	ف
0642	ق
0643	ك
0644	ل
0645	م
0646	ن
0647	ه
0648	و
0649	ى
064A	ي

Appendix IV

A List of Arabic Discourse Markers

Discourse Marker	Meaning	Function	Correlated relation
ايضا	Also	Additive	Elaboration
كما/كما ان	Likewise, furthermore	Additive	Elaboration
اضافة الي/ اضف الي ذلك ان/ يضاف الى ذلك (ان)	In addition (to), moreover	Additive	Elaboration
فضلا عن ان	Besides	Additive	Elaboration
حتى	Even	Additive	Elaboration
كذلك	Likewise, furthermore	Additive	Elaboration
لكن	However, but	Contrastive	Concession
إلا أن	However, but	Contrastive	Concession
غير أن	However, but	Contrastive	Concession
بيد أن	However, but	Contrastive	Concession
ف	Since, for, so, thus	Explanatory	Reason
فقد + الفعل الماضي	Since, for	Explanatory	Reason
إذ أن/إذ	Since, for	Explanatory	Reason
خصوصا أن	Especially that	Explanatory	Reason
أي أن/أي	That is, i.e., in other words	Explanatory	Interpretation
ما /ذلك يعني أن/بمعنى أن يعني أن	This means that	Explanatory	Interpretation
لذلك	Thus, therefore	Inferential	Result
لذا	Thus, therefore	Inferential	Result
لهذا	Thus, therefore	Inferential	Result
من ثم	Thus, therefore	Inferential	Result

Discourse Marker	**Meaning**	**Function**	**Correlated relation**
من هنا	Thus, therefore	Inferential	Result
عليه	Thus, therefore	Inferential	Result
بالتالي	Thus, therefore	Inferential	Result
هكذا	Thus, and so	Inferential	Result
ثم	Then	Sequential	Sequence
وقد + الفعل الماضي	And	Background	Background
الذي لا شك /لا شك في أن فيه هو أن	There is no doubt that, undoubtedly	Expresses certainty	Certainty
من المؤكد /بالتأكيد/الأكيد أن أن	Surely, definitely	Expresses certainty	Certainty
حقيقة/الحقيقة أن	The truth is, truly	Expresses certainty	Certainty
الأرجح أن/من المرجح أن	It is more likely	Expresses certainty	Certainty
الصحيح هو أن	The truth is , the reality is	Expresses certainty	Certainty
واضح أن/من الواضح أن	It is evident that, it is clear that	Expresses certainty	Certainty
في /لقد أثبت الواقع أن في واقع الأمر/الواقع	Reality has proven that, in fact, as a matter of fact	Expresses certainty	Certainty
لقد	Indeed	Expresses certainty	Certainty
غني عن القول أن	It goes without saying	Expresses certainty	Certainty
صار معروفا أن	It became known that	Expresses certainty	Certainty
من الطبيعي أن	Naturally, obviously	Expresses certainty	Certainty
ليس سرا أن/لم يعد سرا أن	It is no longer a secret, it is obvious	Expresses certainty	Certainty
المثير للسخرية أن	What is ironic is that	Evaluative	Evaluation
إنها مأساة فعلا أن	It is truly a tragedy	Evaluative	Evaluation
طبعا	Of course	Evaluative	Evaluation

Discourse Marker	**Meaning**	**Function**	**Correlated relation**
اخيرا	At last	Evaluative	Evaluation
لكأن/كأن	As if	Evaluative	Evaluation
على المرء ان يأخذ في الحسبان أن	It should be taken into consideration that	Evaluative	Evaluation
هذا دليل واضح على أن	This is clear evidence that	guide the interpretation process	Evaluation
السبب بكل بساطة هو أن	The reason simply is that	guide the interpretation process	Reason
يؤشر ذلك إلى أن	This indicates that	guide the interpretation process	Interpretation
أبعد من هذا	Moreover, beyond that	guide the interpretation process	Elaboration
فوق هذا	Moreover, beyond that	guide the interpretation process	Elaboration
النتيجة أن	The result is that	guide the interpretation process	Evaluation
صحيح أن	It is true that	guide the interpretation process	Interpretation
مما /ينبغي التذكير هنا بأن يذكر أن	It is noteworthy here that	guide the interpretation process	Background
أعتقد أن	I think that	introduce writer point of view	View
يظهر أن	It seems that	introduce writer point of view	View
يبدو أن\يبدو	It seems that	introduce writer point of view	View
الملاحظ أن\يلاحظ أن	It is noticed that	introduce writer point of view	View
لذا	Thus, therefore	Inferential/Resultative	Evaluation
من هنا	Thus, therefore	Inferential/Resultative	Evaluation

Discourse Marker	**Meaning**	**Function**	**Correlated relation**
بالتالي	Thus, therefore	Inferential/Resultative	Evaluation
هكذا	Thus, and so	Inferential/Resultative	Evaluation
الواقع أن	As a matter of fact	Evaluative	Evaluation
لا بد أن	It is certain that	Evaluative	Evaluation
المثير للأسى أن	Sadly	Evaluative	Evaluation
لسوء الحظ فإن	Unfortunately	Evaluative	Evaluation
من المستغرب أن	It is surprising that	Evaluative	Evaluation
يبدو أن\يبدو	It seems that	Evaluative	Evaluation
الأمر \المثير في الأمر أن الأمر \المثير للاهتمام هو أن المثير الآخر أن	What is interesting about the matter is that	Attention getting	Evaluation
المفارقة القائمة حاليا هي أن	Ironically	Attention getting	Evaluation
الأكثر أهمية أن	Most importantly	Attention getting	Evaluation
الغريب في الأمر أن	Oddly	Attention getting	Evaluation
اللافت أن	What is interesting is	Attention getting	Evaluation
المشكلة هي أن	The problem is that	Attention getting	Evaluation
لنبادربالإشارة إلى أن	Firstly, it must be mentioned that	Attention getting	Evaluation
صحيح أن	It is true that	Guides interpretation	Evaluation
علينا الاعتراف أن	We should admit that	Appeal to the reader	Evaluation
أخيرا	Lastly	Sequential	Sequence
أولا	Firstly	Sequential	Sequence
ثانيا	Secondly	Sequential	Sequence
ثالثا	Thirdly	Sequential	Sequence
ذاك أن	That is because	Explanatory	Reason
حيث	Since	Explanatory	Reason
بناء على ذلك	According to that	Explanatory	Result

Discourse Marker	**Meaning**	**Function**	**Correlated relation**
بذلك	Thus	Explanatory	Result
من أجل ذلك	Because of that	Explanatory	Result
لأنه كذلك	Because of that	Explanatory	Result
من جهة أخرى	On the other hand	Contrastive	Concession
انما	But	Contrastive	Concession
على أن	But	Contrastive	Concession
على رغم ذلك ف	Despite that	Contrastive	Concession
مع ذلك ف	Despite that	Contrastive	Concession
هذا	This	Additive	Elaboration
يجئ هذا	This comes	Additive	Elaboration
ناهيك عن أن	Moreover	Additive	Elaboration

Appendix V

Arabic Stop Words List

انها	الرغم	امامنا	اي	اولائكم	اياهم	بانه	بايا
اثناء	السابق	امامه	اياه	اولائكما	اياهما	باولئك	باية
اجل	السواء	امامها	ايضا	اولائكن	اياهن	بآخر	بايها
احد	الغير	امامهم	اين	ايا	اياي	باحد	بايهم
احدى	القادم	امامهما	ايها	ايان	بئس	باشياء	بايهما
اخيرا	اللاتي	امامهن	اخر	اية	بالامام	باقل	بايهن
اذ	اللاحق	امامي	ابدا	اينما	بالامر	بالا	باحدى
اذا	اللتان	امس	احيانا	ايها	بالاضافة	بان	باذا
اذن	اللتين	ان	اخرى	ايهن	بالتالي	بانا	بالا
ازاء	اللذان	انا	اخيرا	اطلاقا	بالتاكيد	بانك	باياك
استمرار	اللذين	انت	ازاء	اليك	بالتي	بانكم	باياكم
اصبح	اللواتي	انتم	اشياء	اليكم	بالذي	بانكما	باياكما
اصبحت	المقبل	انتما	اقل	اليكما	بالذين	بانكن	باياكن
اكثر	الممكن	انتن	اكثر	اليكن	بالذين	باننا	باياه
الا	المنصرم	انك	الست	الينا	بالرغم	بانني	باياها
الان	النحو	انكم	الستم	اليه	بالضبط	بانه	باياهم
الامام	الي	انكما	الستما	اليها	بالغير	بانها	باياهن
الامر	اليه	انكن	الستن	اليهم	بالقول	بانهم	بایاي
الاطلاق	اليها	انما	السن	اليهما	بالاتي	بانهما	ببضع
البعض	اليهم	اننا	اليس	اليهن	باللتان	باني	ببضعة
التي	ام	انني	اليست	انا	باللتني	باواخر	ببعض
الجاري	اما	انه	اليسوا	انها	باللذان	باولاء	ببعضها
الحالي	امام	انهم	اني	اياك	باللذين	باولائك	ببعضهم
الخ	امامك	انهما	اواخر	اياكم	باللواتي	باولائك	بتلك
الذان	امامكم	انهن	اولا	اياكن	بالنسبة	باولائكما	بحيث
الذي	امامكما	او	اولاء	اياه	بامكان	باولائكن	بدلا
الذين	امامكن	اولئك	اولائك	اياها	بان	باي	بدون

بدوننا	بعدها	بلى	بينكم	حاليا	خصوصا	دونهم	شيئا
بدونه	بعض	بما	بينكما	حتما	خصيصا	دونهما	شيئان
بدونها	بعضنا	بماذا	بينكن	حتى	خلا	دونهن	شيئين
بدونهم	بعضها	بمتى	بينما	حسب	خلال	ذا	ضدك
بدونهما	بعضهم	بمزيد	بيننا	حوالي	خلاله	ذات	ضدكم
بدونهن	بغض	بمفرد	بينه	حول	خلف	ذاتك	ضدكما
بذا	بغير	بمن	بينها	حولك	خلفك	ذاتكما	ضدكن
بذاك	بغيرك	بن	بينهم	حولكم	خلفكم	ذاته	ضدنا
بذلك	بغيركم	بنا	بينهما	حولكن	خلفكما	ذاتهم	ضده
بذو	بغيركما	بنحو	بينهن	حولنا	خلفكن	ذاتهما	ضدها
بذي	بغيركن	بنسبة	بيني	حوله	خلفنا	ذاتهمن	ضدهم
برغم	بغيرنا	به	تحته	حولها	خلفه	ذاك	ضدهم
بسبب	بغيره	بهؤلاء	تقريبا	حولهم	خلفها	ذلك	ضدهما
بسوى	بغيرها	بها	تقول	حولهن	خلفهم	ذلكم	ضدهن
بشان	بغيرهم	بهاتان	تكن	حولي	خلفهما	ذلكما	ضدي
بشكل	بغيرهما	بهاتين	تكون	حيث	خلفهما	ذو	ضدين
بشئ	بغيرهن	بهذا	تكونوا	حيثما	خلفهن	ذي	ضرورة
بشيئا	بغيري	بهذان	تلك	حين	خلفي	ربما	ضروري
بشيئان	بك	بهذه	تلكم	حينئذ	دائما	رغم	ضروريا
بشيئين	بكافة	بهذي	تلكما	حينا	داخلا	رغما	ضمن
بصورة	بكل	بهذين	تماما	حينذاك	دون	رقم	طالما
بضع	بكم	بهل	ثم	حينما	دونك	سواء	طويل
بضعة	بكما	بهم	ثمة	حينه	دونكم	سوف	طويلا
بعد	بكن	بهما	جدا	حينها	دونكما	سوى	طويلة
بعدئذ	بكيف	بهن	جيدا	خارجا	دوننا	شانه	ظل
بعدة	بل	بين	حاشا	خاصا	دونه	شتى	عام
بعدم	بلا	بينك	حالما	خاصة	دونها	شئ	عامة

عبر	عنده	فاكثر	فانت	فاليها	فبالذي	فبماذا	فتحت
عدا	عندها	فالان	فانتم	فاليهم	فبالذين	فبنا	فتلك
عدة	عندهم	فالتي	فانتما	فاليهما	فبالغير	فبنسبة	فثم
عدم	عندهما	فالذي	فانتن	فاليهن	فبالقول	فبهؤلاء	فجاة
عدمه	عندهن	فالذين	فانه	فاما	فباللاتي	فبها	فحاشا
عديدة	عنك	فالغير	فانهم	فان	فبالاتان	فبهاتان	فحيث
عسى	عنكم	فالقول	فاني	فانا	فباللتين	فبهاتين	فحيثما
على	عنه	فاللاتي	فاولئك	فانك	فباللذين	فبهذا	فحين
عليك	عنها	فاللتان	فاولاء	فانكم	فباللذين	فبهذان	فحينئذ
عليكم	عنهم	فاللتين	فاولائك	فانكما	فباللواتي	فبهذه	فحينا
عليكما	عنهما	فاللذان	فاولائكم	فاننا	فالبنسبة	فبهذين	فحينذاك
عليكن	عنهن	فاللذين	فاولائكما	فانه	فباولئك	فبهم	فحينما
علينا	عني	فاللواتي	فاولائكن	فانها	فباولا	فبهما	فحينها
عليه	غير	فان	فاي	فانهم	فبتلك	فبهن	فخلا
عليها	غيرك	فانك	فايان	فانهما	فبحيث	فبين	فخلال
عليهم	غيركم	فاننا	فاين	فاني	فبذا	فبينك	فدائما
عليهما	غيركما	فانه	فاينما	فاياك	فبذاك	فبينكم	فذا
عليهن	غيركن	فانها	فاذ	فاياكم	فبذلك	فبينكما	فذاك
عما	غيرنا	فانهم	فاذا	فاياكما	فبذي	فبينكن	فذلك
عن	غيره	فاولئك	فالا	فاياكن	فبعد	فبينما	فذو
عنا	غيرها	فاحد	فالى	فاياه	فبعدة	فبيننا	فذي
عند	غيرهم	فاقل	فاليك	فاياها	فبك	فبينه	فسواء
عندئذ	غيرهما	فاكثر	فاليكم	فاياهما	فبكل	فبينها	فسوف
عندك	غيرهن	فالا	فاليكما	فاياهن	فبكم	فبينهم	فسوى
عندكم	غيري	فاما	فاليكن	فاياي	فبكما	فبينهما	فطالما
عندكما	فاذ	فان	فالينا	فبئس	فبكن	فبينهى	فعدا
عندما	فاذا	فانا	فاليه	فبالتي	فبما	فبيني	فعدة

فعدم	فعنكما	ففيك	فكانك	فلاولئك	فلكل	فلهذه	فمدام
فعلا	فعنها	ففيكم	فكانه	فلاحدى	فلكلا	فلهذين	فمدة
فعلى	فعنهم	ففيكن	فكانهم	فلبئس	فلكلتا	فلهم	فمع
فعليك	فعنهما	ففيما	فكانهما	فلتلك	فلكم	فلهما	فمعا
فعليكم	فعنهن	ففينا	فكانهن	فلدي	فلكما	فلهن	فمعك
فعليكما	فعني	ففيه	فكثيرا	فلديك	فلكن	فلو	فمعكم
فعليكن	فغير	ففيها	فكذلك	فلديكم	فلكنك	فلولا	فمعكما
فعلينا	فغيرك	ففيهم	فكل	فلديكما	فلكنهم	فلولاك	فمعكن
فعليه	فغيركم	ففيهما	فكلا	فلدينا	فلكنهما	فلولاكم	فمعنا
فعليها	فغيركما	ففيهن	فكلانا	فلديه	فلكنهن	فلولاكما	فمعها
فعليهم	فغيركن	فقبل	فكلاهما	فلديها	فلكي	فلولاكن	فمعهم
فعليهما	فغيرنا	فقد	فكلتا	فلديهم	فلكيلا	فلولانا	فمعهن
فعليهن	فغيرهم	فقديما	فكلكم	فليديهما	فلم	فلولاها	فمعي
فعن	فغيرهما	فقط	فكلنا	فلديهن	فلما	فلولاهم	فمما
فعنا	فغيرهن	فقلت	فكلها	فلذا	فلماذا	فلولاهما	فمن
فعند	فغيري	فقول	فكلهم	فلذاك	فلمذا	فلولاهن	فمنا
فعندئذ	ففوق	فكالتي	فكلهن	فلذلك	فلن	فلولاي	فمنذ
فعندك	ففوقك	فكالذي	فكلينا	فلذي	فلنا	فليس	فمنك
فعندكم	ففوقكم	فكالذين	فكليهما	فلست	فله	فليست	فمنكم
فعندكما	ففوقكما	فكالقول	فكم	فلستم	فلهؤلاء	فليسوا	فمنكن
فعندما	ففوقكن	فكاللاتي	فكما	فلستما	فلها	فما	فمننا
فعنده	ففوقنا	فكاللتان	فكي	فلستن	فلهاتان	فماذا	فمنها
فعندها	ففوقها	فالكالتين	فكيف	فلسوف	فلهاتين	فماعدا	فمنهم
فعندما	ففوقهم	فكاللذان	فكيلا	فلعدم	فلهتان	فمتى	فمنهما
فعندنى	ففوقهما	فكاللذين	فلا	فلعل	فلهتين	فمثل	فمنهن
فعنك	ففوقهن	فكاللواتي	فلاحد	فلقد	فلهذا	فمثلا	فمني
فعنكم	ففي	فكان	فلانه	فلك	فلهذان	فمثلما	فمهما

فنحن	فوقهم	كاللذين	كاياها	كماذا	لاي	لايهما	لذلك
فهؤلاء	فوقهما	كاللواتي	كاياهم	كمن	لاخر	لايهن	لذو
فهاتان	فوقهن	كان	كاياهما	كن	لاحد	لاحدى	لذي
فهاتين	في	كانا	كاياهن	كنا	لامام	لاياك	لست
فهتان	فيك	كانت	كاياي	كنت	لامامك	لاياكم	لستم
فهتين	فيكم	كانتا	كبيرا	كنتم	لامامكم	لاياكما	لستما
فهذا	فيما	كانوا	كتلك	كنتما	لامامكما	لاياكن	لستن
فهذان	فيها	كاحد	كثيرا	كهؤلاء	لامامكن	لاياه	لسن
فهذه	فيهم	كان	كذا	كهاتين	لامامنا	لاياها	لسوف
فهيذي	فيومئذ	كانك	كذاك	كهذا	لامامها	لاياهم	لعدم
فهذين	قبل	كانكم	كذلك	كهذه	لامامهم	لاياهما	لعل
فهل	قبله	كاننا	كذو	كهذي	لامامهما	لاياهن	لغير
فهم	قبلها	كانها	كسوى	كهذين	لامامهن	لاياي	لقد
فهما	قد	كانهم	كغير	كونه	لامامي	لبئس	لك
فهن	قديما	كانهما	ككل	كونها	لانك	لبعض	لكل
فهنا	قريبا	كانهن	كل	كونوا	لانكم	لتلك	لكلا
فهناك	كافة	كاني	كلا	كي	لانكما	لدي	لكلتا
فهو	كافيا	كاولائك	كلانا	كيف	لانكن	لديك	لكم
فهي	كالان	كاولائكم	كلاهما	كيلا	لاننا	لديكم	لكما
فوق	كالتي	كاولائكما	كلتا	لئلا	لانني	لديكما	لكن
فوقك	كالذي	كاولائكن	كلكم	لا	لانهما	لدينا	لكنك
فوقكم	كالذين	كاي	كلما	لابد	لاواخر	لديها	لكنها
فوقكما	كالقول	كاحدى	كلهن	لان	لاي	لديهم	لكنهم
فوقكن	كاللاتي	كاياك	كلينا	لانه	لايا	لديهما	لكنهما
فوقنا	كاللتان	كاياكم	كليهما	لانها	لاية	لديهن	لكنهن
فوقه	كاللتين	كايكما	كم	لانهم	لايها	لذا	لكني
فوقها	كاللذان	كاياكن	كما	لاولئك	لايهم	لذاك	لكي

لكيلا	لهتان	مؤكدا	معها	هاتان	ورائكن
للامام	لهتين	ما	معهم	هاتين	ورائهم
للامر	لهذا	مادام	معهما	هاذين	ورائهما
للتي	لهذان	ماذا	معهن	هامة	ورائهن
للذي	لهذه	مازال	معي	هانت	يا
للذين	لهذي	مازالت	مما	هانتم	يبدو
للغاية	لهذين	ماعدا	ممكن	هانذا	يكن
للاتي	لهم	ماهو	ممكنا	هذا	يكون
للتان	لهما	متى	ممن	هذان	يكونوا
للتين	لهن	مثل	من	هذه	يلي
للذان	لو	مثلا	منا	هذي	يمكن
للذين	لولا	مثلما	منذ	هذين	يومئذ
للواتي	لولاك	مثلها	منك	هكذا	
للمزيد	لاولاكم	مثلهم	منكم	هل	
لم	لولاكما	مدة	منكما	هم	
لما	لاولاكن	مرة	منكن	هما	
لماذا	لولانا	مزيد	مننا	هن	
لمدة	لولاها	مزيدا	منها	هنا	
لمزيد	لولاهم	مطلقا	منهم	هناك	
لميزيدا	لولاهما	مع	منهما	هنالك	
لمن	لولاهن	معا	منهن	هو	
لن	لولاي	معظم	مني	هي	
لنا	لي	معك	مهما	وراء	
لهؤلاء	ليس	معكم	نحن	وراءه	
لها	ليست	معكما	نظرا	ورائك	
لهاتان	ليسوا	معكن	نعم	ورائكم	
لهاتين	ليكون	معنا	هؤلاء	ورائكما	

References

Abu El-Khair, I. (2006) "Effects of Stop Words Elimination for Arabic Information Retrieval: A comparative Study", *International Journal of Computing & Information Sciences*, Vol. 4 (3), PP: 119-133.

Abu Hilal Al-Askari, A. B. (1952), "Al-Sina'atayn", Essa Al-Halabi press.Egypt.

Akour, M., Abufardeh, S., Magel, K. and Al-Radaideh Q. (2011) "QArabPro: A Rule Based Question Answering System for Reading Comprehension Tests in Arabic", *American Journal of Applied Science 8 (6)*. PP: 652-661.

Alajmi, A., Saad, E.M. and Darwish, R.R. (2012) "Toward an Arabic Stop-words List Generation", *International Journal of Computer Applications*, Vol 46- No.8, PP:8-13.

Al-Fedaghi, S. S. and Al-Anzi, F.S. (1989). "A new algorithm to generate Arabic root-pattern forms", In *Proceedings of the 11th National Computer Conference*, King Fahd University of Petroleum & Minerals, Dhahran, Saudi Arabia, PP: 04-07.

Al-Kohlani, F. A. (2010) "The Function of Discourse Markers in Arabic Newspaper Opinion Articles", *PhD thesis, Georgetown University*, Washington.

Allan, J., Callan, J., Feng, F-F., and Malin D. (1999) "INQUERY and TREC-8", In *Proceedings of the Eighth Text Retrieval Conference, NIST Special Publication 500-246*, Maryland, PP: 637.

Al-Sanie, W. (2005) "Towards an Infrastructure for Arabic Text Summarization Using Rhetorical Structure Theory", *Master's thesis*, King Saud University, Saudi Arabia.

Al-Serhan, H., Al Shalabi, R. and Kannan, G. (2003). "New Approach for Extracting Arabic Roots", In *Proceedings of the 2003 Arab Conference on Information Technology (ACIT 2003)*, Egypt, PP: 42-59.

Al-Shammari, E. and Lin, J (2008) "Towards an Error Free Stemming", In *Proceedings of the 2nd ACM workshop on improving non English web searching*, USA, PP: 9-16.

Altenberg, B. (1984) "Causal Linking in Spoken and Written English". *Studia Linguistica*, Vol .38(1), PP: 20-69.

Al-Zubaydi, M.M. (1888), "Taj Al-Aroos", Al-Haiat. Lebanon.

Aria, K. and Handayani, A. N. (2012) "Question Answering System for an Effective Collaborative Learning", *(IJACSA) International Journal of Advanced Computer Science and Applications*, Vol. 3(1), PP: 60-64.

Asher, N. and Lascarides, A. (2003) "Logics of Conversations", *Cambridge University Press.*

Beesley, K. R. (1996) "Arabic Finite-State Morphological Analysis and Generation", In *Proceedings of the 16th international Conference on Computional Linguistics COLING-96*, Copenhagen, PP: 89-94.

Benajiba, Y., Rosso, P. and Lyhyaoui, A. (2007a) "Implementation of the ArabiQA Question Answering System's Components", In *Proceedings of workshop on Arabic Natural Language Processing, 2nd Information Communication Technologies Int.* Morocco.

Benajiba, Y., Rosso, P. and Soriano, J. M. G. (2007b) "Adapting the JIRS Passage Retrieval System to the Arabic Language", In *Proceeding of 8th International Conference on Computational Linguistics and Intelligent Text Processing*, PP: 530-541.

Bernardi, R., Valentin, J., Gilad, M. and Maarten, D. R. (2003) "Selectively Using Linguistic Resources throughout the Question Qnswering Pipeline". In *Proceedings of the 2nd CoLogNET-ElsNET Symposium.*

Bilotti, M.W. (2004) "Query Expansion Techniques for Question Answering". *Master's thesis*, Massachusetts Institute of Technology.

Blakemore, D. (1996) "Are Apposition Markers Discourse Markers?", *Journal of Linguistics, Cambridge University Press*, Vol. 32, PP: 325-347.

Blakemore, D. (2003) "Discourse and Relevance Theory", *The Handbook of Discourse analysis,* Oxford: Blackwell, PP: 100-115.

Blancol, E., Castell, N. and Moldovan, D. (2008) "Causal Relation Extraction", In *Proceedings of the International of Conference on Language Resources and Evaluation, LREC,* Morocco, PP: 310-313.

Bobrow, D. G., Kaplan, R. M., Kay, M., Norman, D. A., Thomson, H. and Winograd, T. (1977) "GUS, a Frame Driven Dialog System", *Artificial Intelligence*, Vol.8 (2), PP: 155-173.

Bosma, W. (2005) "Query-Based Summarization Using Rhetorical Structure Theory", In *Proceedings of the 15th meeting of Computational Linguistics*, Leiden, PP: 29-44.

Breck, E., Burger, J., Ferro, L., Hous, D., Light, M. and Mani, I. (2000) "Another Sys Called Qanda", In *Proceddings of the Ninth Text REtrieval conference, NIST Special Publication 500-246,* Maryland, PP: 369-379.

Brinton, L. J (1996) "Pragmatic Markers in English: Grammaticalization and Discourse Functions". *Walter de Gruyter*, Berlin.

Burke, R., Hammond, K., Kulyukin, v., Lytimen, S., Tomuro, N., and Schoenberg, S. (1997) "Question Answering from Frequently-Asked Question Files: Experiences with the FAQFinder System". *AI Magazine*, Vol. 18(2), PP: 57-66.

Burstein, J. and Marcu, D. (2003) "A Machine Learning Approach for Identification of Thesis and Conclusion Statements in Student Essays" *Computers and the Humanities*, 37 (4), PP: 455- 467.

Carlson, L., Marcu, D. and Okurwski, M. E. (2001) "Building a Discourse-Tagged Corpus in the Framework of Rhetorical Structure Theory", In *Proceeding of the Second SIGdial Workshop on Discourse and Dialogue*, Denmark, Vol.16, PP: 1-10.

Carlson, L., Marcu, D., Okurowski, M.E. (2002) RST discourse Treebank, *linguistic Data Consortium*, Available online https://catalog.ldc.upenn.edu/LDC2002T07, Accessed Auguest, 2014.

Choi, K., Pacana, R.M., Tan, A. L., Yiu, J. and Lim, N. R. (2011) "A Question Answering System that Performs Evaluations and Comparisons on Structured Data for Business Intelligence in Biotechnology" *Uncertainty Reasoning and Knowledge Engineering (URKE)*, Vol.1, PP: 137-140.

Cooper, R. J. and Ruger, S. M. (2000) "A Simple Question Answering System", In *proceedings of the ninth Text REtrieval Conference, NIST Special Publication 500-249,* Maryland, PP: 249.

Cormack, G. V., Clarke, C. L. A., Palmer, C. R. and Kishman, D. I. E. (1999) "Fast Automatic Passage Ranking (MultiText Experiments for TREC-8)", In *Proceedings of the Eighth Text Retrieval Conference, NIST Special Publication 500-246,* Maryland, PP: 735.

Corston-Oliver, S. (1998) "Computing Representations of the structure of written Discourse", *PhD thesis, University of Califorinia,* Santa Barbara.

Dang, H. T., Kelly, D. and Lin, J. J. (2007) "Overview of the TREC 2007 Question Answering Track*",* In *Proceedings of the sixteenth Text REtrieval Conference, NIST Special Publication 500-274, Maryland,* PP: 113.

Dang, H. T., Lin, J. and Kelly, D. (2006) "Overview of the TREC 2006 Question Answering Track*",* In *Proceedings of the Fifteenth Text REtreival Conference, NIST Special Publication 500-272,* Maryland, PP: 99.

Duwairi, R. M. (2005) "A Distance-based Classifier for Arabic Text Categorization", In *Proceedings of the 2005 International Conference on Data Minig,* USA.

El Kourdi, M., Bensaid, A. and Rachidi, T. (2004) "Automatic Arabic Document Categorization Based on the Naive Bayes Algorithm", In *Procceding of the Workshop on Computational Approaches to Arabic Script-based Languages*, USA. PP: 51-58.

Fareh, S. H., and Hamdan, J. (1999) "The Translation of Arabic 'Wa' into English: Some Problems and Implications", *Human and Social Science,* Jordan.

Feldman, R. and Sanger, J. (2007) "The Text Mining Handbook", *Cambridge University press,* UK.

Feng, V. W. and Hirst, G. (2012). "Text-level Discourse Parsing with Rich Linguistic Features", In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-2012),* Jeju, Korea, PP: 60-68.

Fraser, B. (1996) "Pragmatic Markers", *Pragmatics,* vol 6 (2), PP:167-190.

Fraser, B. (1999). "What are DMs?", *Journal of Pragmatics,* Vol.31, PP: 931-925.

Fukumoto, J. (2007) "Question Answering System for Non-factoid Type Questions and Automatic Evaluation Based on BE Method", In *Proceedings of NTCIR-6 Workshop Meeting*, Tokyo, PP: 441-447.

Gaizauskas, R. and Wilks, Y. (1997) "Information Extraction: Beyond Document Retrieval", *Computaltioal Linguistics and Chinese Language Processing,* Vol. 3 (2), PP: 17-60.

Garcia, D. (1997) "COATIS, an NLP System to Locate Expressions of Actions Connected by Causality Links", In *Proceedings of the 10th European Workshop on Knowledge Acquisition, Modeling and Management*, PP: 347-352.

George Weber. (1997) "The World's 10 most influential Languages", *Language Today,* Vol.2.

Ghorbel, H., Ballim, A. and Coray, G. (2001) "ROSETTA: Rhetorical and Semantic Environment for Text Alignment", In *Proceedings of Corpus Linguistics 2001*, Lancaster, UK, PP: 224-233.

Girju, R. and Moldovan, D. (2002) "Mining answers for causation questions", In *Proceedings of the American Association for Artificial Intelligence (AAAI)-Spring Symposium*, PP: 15-25.

Gonsalves, R. J. (1986) "A Decompositional Analysis of Causative Verbs", CUNY forum, 12, PP: 31-65.

Green, B. F., Wolf, A. K., Chomsky, C. and Laugherty, K. (1961) "BASEBALL: An Automatic Question Answerer", In *Proceedings of the Western joint computer conference*, PP: 219-224.

Greenwood, M. A. (2004) "AnswerFinder: Question Answering from your Desktop", In *Proceedings of the 7th Annual Colloquium for the UK Special Interest Group for Computational Linguistics*, Brimingham, PP: 75-80.

Greenwood, M. A. and Saggion, H. (2004) "A pattern Based Approach to Answering Factoid, List and Definition Questions", In *Proceedings of the 7th RIAO Conference*, France, PP: 232-243.

Grosz, B. J and Sidner, C. L. (1986) "Attention, Intentions, and the Structure of Discourse", *Computational Linguistics*, USA, Vol.12 (3), PP: 175-204.

Halliday, M.A.K. and Hasan, R. (1976) "Cohesion in English", *Longman*. London.

Hammou, B., Abu-Salem, H. and Lytinen, S. (2002) "QARAB: A Question Answering System to Support the Arabic Language", In *Proceeding of the 40th Association for Computational Linguistics on Computational Approaches to Semitic Languages, ACL*, USA, PP: 55-65.

Harabagiu, S., Moldovan, D., Clark, C., Bowden, M., Hickl, A., and Wang, P. (2005) "Employing Two Question Answering Systems in TREC-2005", In *Proceedings of the Fourteenth Text REtrieval Conference, NIST Special Publication,* Maryland, PP:482.

Harabagiu, S., Moldovan, D., Clark, C., Bowden, M., Williams, J., and Bensley, J. (2003) "Answer Mining by Combining Extraction Techniques with Abductive Reasoning", In *Proceedings of the Twelfth Text REtrieval conference, NIST Special Publication 500-255*, Maryland, PP:375.

Harabagiu, S., Moldovan, D., Pasca, M., Mihalca, R., Surdeanu, M., Bunescu, R., Girju, R., Rus, B., and Morarescu, P. (2000) "FALCON: Boosting Knowledge for Answer Engines", *Proceedings of the ninth Text REtrieval Conference, NIST Special Publication 500-249*, Maryland, PP: 479.

Harabagiu, S., Moldovan, D., Pasca, M., Mihalcea, R., Surdeanu, M., Bunescu, R., Girju, R., Rus, V., and Morarescu, P. (2001) "The role of lexico-Semantic Feedback in Open-Domain Textual Question-Answering". In *proceedings of the 39th Meeting of the Association for Computational Linguistics (ACL)*, PP: 274.

Haskkour, N. (1990) "Al-Sababia fe Rarkeeb Aljomla Alarabya", *Master's theise*, Aleppo University, Syria.

Haskkour, N. (2009) "Al-Adwat Al-Nahawya fe Mughne Al-Labib Lebn Hesham", *Dar AL-FURQAN*, Syria.

Hatim, B. (1998) "Communication Across Cultures: Translation Theory and Contrastive Text Linguistics", *University of Exeter Press*, UK.

Hernault, H., Prendinger, H., duVerle, D. and Ishizuka, M. (2010). "HILDA: A Discourse Parser Using Support Vector Machine Classification", *Dialogue and Discourse*, Vol. 1 (3), PP: 1-33.

Higashinaka, R. and Isozaki, H. (2008) "Corpus-based Question Answering for why-Question", In *Proceedings of the 3rd International Joint Conference on Natural Language Processing*, India, vol.1, PP: 418-425.

Hobbs, J. R. (1985) "On the Coherence and Structure of Discourse", *Technical Report CSLI-85-37, Center for the Study of Language and Information*, Stanford University.

Ibn Jinni, A. O. (1952) "Al-Khasaes", Dar Al-Huda press. Lebanon.

Itto, A. and Bouma, G. (2011) "Extracting Explicit and Implicit Causal Relations from Sparse, Domain-Specific Texts", In *Proceedings of the 16th International Conference on Applications of Natural Language to Information Systems, NLDB 2011*, PP: 52-63, Spain.

Ittycheriah, A. and Roukos, S. (2002) "IBM's Statistical Question Answering System- TREC 11" In *Proceedings of the Eleventh Text REtrieval Conference, NIST Special Publication 500-251*, Maryland, PP: 229.

Ittycheriah, A., Franz, M., Zhu, W. and Ratnaparkhi, A. (2000) "IBM's Statistical Question Answering System", In *Proceedings of the ninth Text REtrieval Conference, NIST Special Publication 500-249*, Maryland, PP: 229.

Jacobs, P. S. and Rau, L. F. (1990) "SCISOR: Extracting Information from On-line News". *Communications of the ACM*, Vol. 33(11), PP: 88-97.

Jattal, M. (1979) "Nezam al-Jumlah", Aleppo University press, Syria, PP: 127-140.

Jones, K. S. (1972) "A statistical interpretation of term specificity and its application in retrieval", *Journal of Documentaton*, Vol. 28(1), PP: 11-21.

Kadri Y. and Benyamina A. (1992) "A Syntax Semantic Analyzer for Arabic Language", *Engineer thesis*, University of Oran.

Kamalski, J., Sanders, T., and Lentz, L. (2008) "Coherence Marking, Prior Knowledge, and Comprehension of Informative and Persuasive Texts: Sorting Things Out", *Discourse Processes*, Vol.45, PP: 323-345.

Kammensjo, H. (2010) "Discourse Connectives in Arabic Lecturing Monologue", *Journal of Near Eastern Studies*, Vol. 69(1) PP: 143-144.

Kanaan, G., Hammouri, A., Al-Shalabi, R. and Swalha, M. (2009) "A New Question Answering System for Arabic Language", *American Journal of Applied Science*, Vol. 6 (4) PP: 795-805.

Katz, B. (1997) "Form Sentence Processing to Information Access on the World Wide Web", In *Proceedings of the American Association for Artificial Intelligence Conference, Spring Symposium, NLP for WWW*, PP: 77-86.

Khoja, S. and Garside, R. (1999) "Stemming Arabic Text**"**, *Computing Department, Lancaster University*, Lancaster, UK.

Khoo, C. S. G., Kornfilt, J., Oddy, R. N. and Myaeng, S. H. (1998) "Automatic Extraction of Cause-Effect Information from Newspaper Text without Knowledge-based Inferencing", *Literary and Linguistic Computing*, Vol.13 (4), PP: 177-186.

Khoo, C.S.G., Chan, S. and Niu, Y. (2000) "Extraction Causal Knowledge from a Medical Database Using Graphical Patterns", In *Proceedings of 38th Annual Meeting of the ACL*, HongKong, PP: 336-343.

Kupice, j. (1999) "MURAX: Finding and Organizing Answers form Text Search". *Natural Language Information Retrieval*, Netherlands, PP: 311-332.

Kupiec, J. (1993) "MURAX: A Robust Linguistic Approach for Question Answering Using an On-line Encyclopedia". *In Proceedings of the 16th Annual Int. ACM SIGR Conference*, PP: 181-190.

Larkey, L. S., Ballesteros, L and Connell M. E**.** (2002) "Improving Stemming for Arabic Information Retrieval: Light Stemming and Co-occurrence Analysis", In *Proceedings of the 25th annual International Conference on Research and Development in Information Retrieval (SIGIR 2002)*, Finland, PP: 275-282.

Lehnert, W. (1977) "The Process of Question Answering", *Doctoral Dissertation*, Yale University, USA.

Lenk, U. (1998) "Marking Discourse Coherence: Functions of Discourse Markers in Spoken English". *Gunter Narr Verlag*, Germany.

Lin, C. J and Chen, H. H. (1999) "Description of Preliminary Results to TREC-8 QA Task", In *Proceedings of the Eighth Text Retrieval Conference, NIST Special Publication 500-246*, Maryland, PP:507.

Lindley, C. A., Davis, J. R., Nack, F. and Rutledge, L. W. (2001) "The Application of Rhetorical Structure Theory to Interactive News Program Generation from Digital Archives", *Technical Report No. INS-R0101, Centrum voor Wiskunde en Informatica*, Netherlands.

Litkowski, K. C. (2000) "Syntactic Clues and Lexial Resources in Question-Answering" In *Proceedings of the ninth Text REtrieval Conference, NIST Special Publication 500-249*, Maryland, PP: 157-166.

Mann, W. C. and Taboada, M (2005) "Introduction to Rhetorical Structure Theory", Available online http://www.sfu.ca/rst, Accessed Augest, 2014.

Mann, W. C. and Taboada, M. (2006) "Rhetorical Structure Theory: Looking back and moving ahead", *SAGE. Discourse Studies*. Vol. 8, PP: 423-459.

Mann, W. C. and Thompson, S. A. (1988) "A Rhetorical Structure Theory: Toward a functional theory of text organization", Text- Interdisciplinary Journal for the Study of Discourse, Vol. 8 (3), PP: 243-281.

Mann, W. C., Matthiessen, C., Thompson S. A. (1993) "Discourse Description: Diverse inguistics analyses of a fund-raising text", *John Benjamin's publishing*, USA.

Mann, W. C., Mtthiessen, C. and Thompson, S. A. (1992) "Rhetorical Structure Theory and Text Analysis", In *A Frame Work for the Analysis of Texts*, PP. 79-195.

Marcu, D. (1996) "Building Up Rhetorical Structure Trees", In *Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI),* Vol 2, PP: 1069-1074.

Marcu, D. (1997) "The Rhetorical Parsing Summarization, and Generation of Natural Language Texts", *PhD Thesis, Department of Computer Science, University of Toronto.* Canada.

Marcu, D. (2000a) "The Rhetorical Parsing of Unrestricted Texts: A surface-based approach", *Computational Linguistics* Vol. 26(3), PP: 395-448.

Marcu, D. (2000b) "The Theory and Practice of Discourse Parsing and Summarization", *MIT Press London,* England.

Marcu, D., Amorrortu, E. and Romera, M. (1999) "Experiments in constructing a corpus of discourse trees". In *Proceedings of the ACL Workshop on Standards and Tools for Discourse Tagging,* College Park, MD, PP: 48-57.

Mathkour, H., Touir, A. and Al-Sanea, W. (2008) "Parsing Arabic Texts Using Rhetorical Structure Theory", *Journal of Computer Science,* Vol. 4 (9), PP: 713-720.

McNamara, D. S. and Kintsch, W. (1996) "Learning from Texts: effects of prior knowledge and text coherence". *Discourse processes,* Vol. 22 (3), PP: 247-288.

Mohammed, F. A., Nasser, K, and Harb, H. M. (1993) "A knowledge Based Arabic Question Answering System (AQAS)", *ACM SIGART Bulletin,* New York, PP: 21-33.

Moldovan, D., Bowden, M. and Tatu, M. (2006) "A Temporally-Enhanced PowerAswer in TREC 2006", In *Proceedings of the Fifteenth Text REtrieval Conference, NIST Special Publication 500-272,* Maryland.

Moldovan, D., Clark, C., and Bowden, M. (2007) "Lymba's PowerAnswer 4 in TREC 2007", In *Proceedings of the sixteenth Text REtrieval Conference, NIST Special Publication 500-274,* Maryland.

Moldovan, D., Harabagiu, S., Girju, R., Morarescu, P., Lascatusu, F., Novischi, A., Badulescu, A. and Bolohan, O. (2002) "LCC Tools for Question Answering", In *Proceedings of the Eleventh Text REtrieval Conference, NIST Special Publication 500-251,* Maryland, PP: 386- 395.

Moldovan, D., Harabagiu, S., Pasca, M., Mihalcea, R., Girju, R., Goodrum, R. and Rus, V. (2000) "The Structure and Performance of an Open-Domain Question Answering System", In *Proceedings of the Conference of the Association for Computational Linguistics,* ACL, PP: 563-569.

Moldovan, D., Harabagiu, S., Pasca, M., Mihalcea, R., Goodrum, R., Girju, R. and Rus, V. (1999) "LASSO: A Tool for Surfing the Answer Net" In *Proceedings of the Eighth Text Reterival Conference, NIST Special Publication 500-246,* Maryland, PP: 175-183.

Mori, T., Ohta, T., Fujihata, K. and Kumon, R. (2003) "A* Search Algorithm for Question Answering", In *Proceedings of the Third NTCIR Workshop*.

Mori, T., Sato, M., Ishioroshi, M., Nakano, Y. N. S. and Kimura, K. (2007) "A Monolithic Approach and a Type-by-Type Approach for Non-Factoid Question-answering", In *Proceedings of NTCIR-6 Workshop Meeting*, Tokyo, PP: 469-476.

Murata, M., Utiyama, M. and Isahara, H. (2000) "Question Answering System Using Similarity-guided Reasoning", *SIG Notes, Information Processing Society of Japan*, PP:181-188.

Niu, Y. and Hirst, G. (2009) "Analyzing the Text of Clinical Literature for Question Answering". In *Violaine Prince and Mathieu Roche, information Retrieval in Biomedicine, IGI Global*, Hershey, PP: 190-220.

Ogden, B., Cowie, J., Ludovik, E., Molina-Salgado, H., Nirenburg, S., Sharples, N. and Sheremtyeva, S. (1999) "CRL's TREC-8 Systems Cross-Lingual IR, and Q&A", In *Proceedings of the Eighth Text Retrieval Conference, NIST Special Publication 500-246*, Maryland, PP: 513-521.

Otair, M. A. (2013) "Comparative Analysis of Arabic Stemming Algorithms", *International Journal of Managing Information Technology*, Vol. 5 (2).

Parsed, R. and Josh, A. (2008) "A Discourse-based Approach to Generating Why-Questions from Texts", In *Proceedings of the Workshop on the Question Generating Shared Task and Evaluation Challenge*, Arlington.

Pasca, M. A. and Harabagiu, S. M (2001) "High performance question/answering", In *Proceedings of the 24th International Conference on Research and Development in Information Retrieval*, Louisiana, USA, PP: 366-374.

Quarteroni, S., Moschitt, A., Manandhar, S. and Basili, R. (2007) "Advanced Structural Representations for Question Classification and Answer Re-ranking", In *Proceedings of ECIR Springer*, Germany, PP: 234-245.

Ravichandran, D. and Hovy, E. (2002) "Learning Surface Text Patterns for a Question Answering System", In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Pennsylvania, PP: 41-47.

Reinhart, T. (1980) "Conditions for Text Coherence", *Poetics Today, Duke University Press*, Vol. 1 (4), PP: 161-80.

Saeed A. T. and Fareh, S. (2006) "Difficulties Encountered by Bilingual Arab Learners in Translating Arabic 'fa' into English", *The International Journal of Bilingual Education and Bilingualism*, Vol. 9(1), PP: 19-32.

Saggion, H., Gaizauskas, R., Hepple, M., Roberts, I. and Greenwood, A. (2004) "Exploring the Performance of Boolean Retrieval Strategies for Open Domain Question Answering", In *Proceedings of the SIGIR Workshop on Information Retrieval for Question Answering (IR4QA)*, UK, PP: 45-52.

Sanders, T. J. and Noordman, L. G. M. (2000) "The role of Coherence Relations and Their Linguistic Markers in Text Processing". *Discourse processes*, Vol. 29 (1), PP: 37-60.

Sarig, L. (1995) "Discourse Markers in Contemporary Arabic". *Zeitschrift fur Arabische Linguistik*, PP: 7-21.

Sawalha, M. and Atwell, E. (2008) "Comparative Evaluation of Arabic Language Morphological Analysers and Stemmers", In *Coling 2008*, Manchester, PP: 107–110.

Schank, R. C. and Abelson, R. P. (1977) "Scripts, Plans, Goals and Understanding: An inquiry into human knowledge structures", *Psychology press*.

Schiffrin, D. (1987), "Discourse markers", *Cambridge University Press*, UK.

Schiffrin, D., Tannen, D. and Hamilton, H. E. (2001) "Discourse Markers: Language, Meaning, and Context", *Basil Blackwell*, Oxford.

Schneuwly, B. (1997) "Textual Organizers and Text Types: Ontogenetic aspects in writing", *Processing Interclausal Relationships: Studies in the production ad comprehension of text*, PP: 245-263.

Segal, E. M., Duchan, J. F. and Scott, P.J. (1991) "The Role of Interclausal Connectives in Narrative Structuring: Evidence from adults' interpretation of simple stories". *Discourse processes*, Vol. 14, PP: 27-54.

Shima, H. and Mitamura, T. (2007) "JAVELIN III: Answering non-factoid Question in Japanese", In *Proceedings of NTCIR-6 Workshop Meeting*, Tokyo, PP: 464-468.

Small, S., Strazalkowski, T., Liu, T., Ryan, S., Salkin, R., Shimizu, N., Kantor, P., Kelly, D., Rittman, R. and Wacholder N. (2004) "HITIQA: Towards analytical question answering", In *Proceedings of 20th International Conference on computational Linguistic*, Switherland.

Soriano, J. M. G., Gomez, M. M. Y., Arnal, E. S. and Rosso, P. (2005) "A Passage Retrieval System for Multilingual Question Answering", In *Proceedings of the 8th International Conference of Text, Speech and Dialogue 2005 (TSD'05)*, Czech Republic, Vol 3658, PP: 443-450.

Soricut, R. Marcu, D. (2003) "Sentence Level Discourse Parsing using Syntactic and Lexical Information", In *Proceedings of the Human Language Technology and North American Association for Computational Linguistics*, Canda. Vol. 1, PP: 149-156.

Soubbotin, M. M. (2001) "Patterns of Potential Answer Expressions as Clues to the Right Answers", In *Proceedings of the tenth Text REtrieval conference, NIST Special Publication 500-250*, Maryland, PP: 293-302.

Srihari R, and Li, W. (1999) "Information Extraction Supported Question Answering", In *Proceedings of the Eighth Text Retrieval Conference, NIST Special Publication*, Maryland, PP: 185-196.

Surdeanu, M., Ciaramita, M. and Zaragoza, H. (2008) "Learning to Rank Answers on Large Online QA Collections", In *Proceedings of ACL-08*, USA, PP: 719-727.

Taboada, M. and Mann, W. C. (2006) "Applications of Rhetorical Structure Theory", *Discourse Studies*, Vol. 8 (4), PP: 567-588.

Takaki, T. (1999) "NTT DATA: Overview of System Approach at TREC-8 ad-hoc and Question Answering". In *Proceedings of the Eighth Text Retrieval Conference, NIST Special Publication 500-246*, Maryland, PP: 523-530.

The Arab Knowledge Report (2011), *United Nations Development Program (UNDP)*, Dubai.

Theijssen, D., Halteren, H. V., Vervrne S. and Boves, L. (2008) "Features for Automatic Discourse Analysis of Paragraph", In *Proceedings of the 18th Meeting of Computational linguistics in the Netherlands 2007*, PP: 53-68.

Timmermn, S. (2007) "Automatic Recognition of Structural Relations in Dutch Text", *Master Thesis, University of Twente*, the Netherlands.

Verberne, S. (2007) "Paragraph Retrieval for Why-question Answering", In *Proceedings of the 30th Annual International ACM SIGR Conference on Research and Development in Information Retrieval*, New York, PP: 922-927.

Verberne, S. (2010) "In Search of the Why: Developing a system for answering why-question". *PhD thesis, Radboud University*, the Netherland.

Verberne, S., Boes, L., Oostdijk, N. and Coppen, P. (2010) "What is not in the Bag of Words for Why-QA?", *Computational Linguistics journal, MIT Press* Vol. 36 (2), PP: 229-245.

Verberne, S., Boves, L., Coppen, P. and Osstdijk, N. (2007) "Discourse-based Answering of Why-questions", *Traitement Automatiques des Langues, Special Issue on Computational Approaches to Document and Discourse*, Vol. 47 (2), PP: 21-41.

Verberne, S., Raaijmakers, S. and Theijssen, D. (2009) "Learning to Rank Answers to Why-Questions", In *Proceedings of the 9th Dutch-Belgian Information Retrieval Workshop (DIR)*, Enschede, PP: 34-41.

Voorhees, E. M. (2000) "Overview of the TREC-9 Question Answering Track", In *Proceedings of the ninth Text Retrieval Conference, NIST Special Publication 500-249*, Maryland, pp: 71-81.

Voorhees, E. M. (2001) "Overview of the TREC 2001 Question Answering Track", In *Proceedings of the tenth Text REtrieval Conference, NIST Special Publication 500-250*, Maryland, PP: 42-51.

Voorhees, E. M. (2002a) "Overview of the TREC 2002 Question Answering Track", *Proceedings of the Eleventh Text REtrieval Conference, NIST Special Publication 500-251*, Maryland, PP: 56-67.

Voorhees, E. M. (2002b) "Overview of TREC 2002", In *Proceedings of the Eleventh text REtrieval Conference, NIST Special Publication 500-251*, Maryland, PP: 1-15.

Voorhees, E. M. (2003a) "Overview of the TREC 2003 Question Answering Track", In *Proceedings of the Twelfth Text REtrieval Conference, NIST Special Publication 500-255*, Maryland, PP: 54-68.

Voorhees, E. M. (2003b) "Overview of TREC 2003", In *Proceedings of the Twelfth Text REtrieval Conference, NIST Special Publication 500-255*, Maryland, PP:1-13.

Voorhees, E. M. (2004a) "Overview of the TREC 2004 Question Answering Track", In *Proceedings of the Thirteenth Text REtrieval Conference, NIST Special Publication 500-261*, Maryland, PP: 58-68.

Voorhees, E. M. (2004b) "Overview of TREC 2004", In *Proceedings of the Thirteenth Text REtrieval Conference, NIST Special Publication 500-261*, Maryland, PP: 1-12.

Voorhees, E. M. (2005) "Overview of TREC 2005", In *Proceedings of the Fourteenth Text REtrieval Conference, NIST Special Publication 500-266*, Maryland, PP:1-15.

Voorhees, E. M. (2006) "Overview of TREC 2006", In *Proceedings of the fifteenth Text REtreival Conference, NIST Special Publication 500-272*, Maryland, PP: 1-16.

Voorhees, E. M. (2007) "Overview of TREC 2007", *Proceedings of the sixteenth Text REtrieval Conference, NIST Special Publication 500-274*, Maryland, PP: 1-16.

Voorhees, E. M. and Dang H. T. (2005) "Overview of the TREC 2005 Question Answering Track", In *Proceedings of the Fourteenth Text REtrieval Conference, NIST Special Publication 500-266*, Maryland, PP: 65-76.

Voorhees, E. M. and Harman, D. (2000) "Overview of the Ninth Text Retrieval Conference (TREC-9)", In *Proceedings of the ninth Text REtieval Conference, NIST Special Publication 500-249*, Maryland, PP:1-13.

Voorhees, E. M. and Harman, D. (2001) "Overview of TREC 2001", In *Proceedings of the tenth Text REtrieval Conference, NIST Special Publication 500-250*, Maryland, PP: 1-15.

Voorhees, E. M. and Tice, D. M. (1999) "The TREC-8 Question Answering Track Evaluation", In *Proceedings of the 8th Text Retrieval Conference (TREC-8), NIST Special Publication*.

Voorhees, E. M. and Tice, D. M. (2000) "Building a Question Answering Test Collection". In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Wang, Y. F. and Petrina, S. (2013) "Using Learning Analytics to Understand the Design of an Intelligent Language Tutor-Chatbot Lucy". *(IJACSA) International Journal of Advanced Computer Science and Applications*, Vol.4 (11), PP: 124-131.

Weizenbaum, j. (1966) "ELIZA – A Computer Program for the Study of Natural Language Communication between Man and Machine", *Communication of the ACM*, Vol. 9 (1), PP: 36-45.

Wilensky, R. (1982) "Talking to UNIX in English: An overview of an On-line UNIX consultant", *Tech rep, University of California*, USA.

Winogra, T. (1972) "Understanding Natural Language", *Academic Press*, New York.

Wolf, F. and Gibson, E. (2003) "A Response to Marcu (2003). Discourse Structure: trees or graphs", Available online http://www.isi.edu/~marcu/discourse/Discourse%20structures.htm, Accesssed April 2011.

Wolf, F. and Gibson, E. (2005) "Representing Discourse Coherence: A Corpus-Based Study", *Computational Linguistics*, Vol. 31 (2), PP: 249-287.

Woods, W. A., Kaplan, R. M. and Nash-Wbber, B. (1972) "The Lunar Science Natural Language Information System: Final Report, Volume 1", *Bolt Beranek and Newman Inc*.

Wright, W. and Caspari, C. P. (1896) "A Grammar of the Arabic Language", *Cambridge University Press*. Uk.

Yang Z., Li Y., Cai J and Nyberg, E. (2014) "QUADS: Question Answering for Decision Support", In *proceedings of SIGR 2014: the Thirty-seventh Annual Internations ACM SIGIR Conference on Research and Development in Information Retrieval*, USA, PP: 375-384

Zaki, M. (2011) "The Semantics and Pragmatics of Demonstratives in English and Arabic", *PhD thesis, University of Middlesex*. UK.