Towards an Infrastructure For Arabic Text Summarization Using Rhetorical Structure Theory

by

Waleed Al-Sanie

A thesis presented to King Saud University in fulfillment of the thesis requirement for the degree of Master in Computer Science

Riyadh, Kingdom of Saudi Arabia

Copyright © Waleed Al-Sanie, 2005
In the name of Allah the most gracious, the most merciful
I authorize King Saud University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize King Saud University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

*Waleed Al-Sanie*
Abstract

Rhetorical structure theory (RST) has been a subject of research since two decades. The theory concerns rhetorical interrelations between the text spans. It attempts to extract the semantic behind the parsed text. Since known, rhetorical structure theory has been studied on several languages. The employment of this theory on Arabic language has never been addressed. This thesis was the answer of this issue; it attempts to set a framework of applying the RST on the Arabic language.

Different applications have been proven to give good results when they use RST as a base. One of such applications is the text summarization. There is no doubt about the importance of the automatic text summarization. It is one of the subjects that attracted researches over the past 20 years. The demand for a fast summary is increasing; businessmen need fast summaries for the reports and documents they have to gain time, researchers need a tool to generate fast summaries for the references to decide whether it is useful or not, and many people with different needs need such application. Because of this, this thesis attempts to develop an infrastructure for Arabic automatic-summarization. The infrastructure is based on RST. It suggests different techniques, algorithms, and design patterns to be considered when dealing with such application.
Acknowledgments

First of all, I thank Allah for everything and in any situation. All thanks go to Allah for all the knowledge and education I gain which leads to the achievement of this thesis.

Second, I would like thank my parents Abdullah and Sara for their encouragement, support and advices in all my life stages; I wouldn't give them what they deserve even if I wrote thousands of papers; all what I can say is that you are participants in all the successes I make in my life. The patience and help of my wife Nouf are not forgotten; I want just to tell her that you showed a wonderful way of what the supportive wife should be. I thank my brothers and sisters for their encouragement and advices.

It is my pleasure to be supervised by Dr.Amir Touir and Dr.Hassan Mathkour. Their patience was unexpected; they gave my invaluable advices and directions especially in the research skills.

There are people who influence others by their experience and great morals. One of those is my previous director in the R&D of advanced electronics company Mohammed Al-Rasheed. I thank him for his sincere recommendations and support which improved my technical skills.

Thanks to all member of the faculty of computer science department. Thanks to any one who participated in the achievement of this thesis either directly or indirectly.
Contents

Chapter 1: Introduction ................................................................................................... 1
  1 Motivation.................................................................................................................. 1
  2 Text Summarization Overview .............................................................................. 3
    2.1 Sentences Selection Techniques ..................................................................... 4
    2.2 Proposed Condensation Processes .................................................................. 5
  3 Thesis Overview ..................................................................................................... 11

Chapter 2: Rhetorical Structure Theory ...................................................................... 12
  1 Introduction........................................................................................................... 12
    1.1 Rhetorical Relations...................................................................................... 13
    1.2 RST Schema ................................................................................................. 16
    1.3 Nuclearity...................................................................................................... 17
    1.4 RST Analysis ................................................................................................ 18
  2 Building the Rhetorical Structure Tree (RS-tree) ................................................. 20
    2.1 RS-tree formalization.................................................................................... 20
    2.2 Building the RS-tree ..................................................................................... 22
  3 RS-tree and Text Summarization .......................................................................... 26

Chapter 3: Arabic Text Summarization Theory ............................................................ 28
  1 Arabic Rhetorical Relations .................................................................................. 28
    1.1 Rhetorical Relations Foundation ................................................................... 29
    1.2 Relations Signaling ....................................................................................... 34
  2 Rhetorical Parsing of Arabic Text ........................................................................ 36
    2.1 Elementary Units Determination .................................................................. 36
    2.2 Relations Classification ................................................................................ 38
    2.3 Text Coherency ............................................................................................. 39
    2.4 The Most Suitable RS-tree for Arabic Text Summarization ...................... 42

Chapter 4: A Pattern-Oriented System Model ............................................................ 49
  1 System Aspects and Requirements ...................................................................... 50
# List of Tables

Tab 1. 1. Condensation rules identified in (Rush et al., 1971). .......................................... 6  
Tab 1. 2. Condensation rules identified in (Mathis et al., 1973). ........................................... 7  
Tab 1. 3. Condensation process identified in (Maybury, 1995). ............................................. 7  
Tab 1. 4. Rephrasing operations identified in (Jing & McKeown, 2000). ........................... 8  
Tab 1. 5. Rephrasing transformation identified in (Saggion, 2000). ................................. 10  
Tab 2. 1. The definition of the evidence relation in (Mann et al., 1992). .............................. 14  
Tab 2. 2. English rhetorical relations as stated in (Mann & Matthiessen, 1991). .............. 15  
Tab 3. 1. Arabic rhetorical relations. ................................................................................ 34  
Tab 3. 2. Example of the features of the cue phrases ....................................................... 36  
Tab 5. 1. Result of auto-summarization software for small size articles. ......................... 69  
Tab 5. 2. Result of auto-summarization software for medium size articles. ..................... 69  
Tab 6. 1. Cue phrases used in the experiment. ................................................................... 72
List of Figures

Fig 1. 1. Automatics summarization process................................................................. 4
Fig 1. 2. Thesis process................................................................................................. 11
Fig 2. 1. Example of five schemas............................................................................... 16
Fig 2. 2. RST analysis for the example 2.1................................................................. 19
Fig 2. 3. Example of the feature status, type, and promotion for an evidence relation that connects two leaf spans.............................................................................. 21
Fig 2. 4. Algorithm that derive all the RS-trees for a given text................................. 26
Fig 3. 1. The process of extracting the Arabic relations from English relations............ 30
Fig 3. 2. Arabic word stemming algorithm................................................................. 42
Fig 3. 3. Two RS-trees generated for one text............................................................. 43
Fig 3. 4. Two summarized texts generated from two RS-trees................................. 43
Fig 3. 5. Arabic writing style observed from the Arabic corpus study......................... 44
Fig 3. 6. The three categories of the RS-tree............................................................... 45
Fig 3. 7. Rhetorical parsing algorithm......................................................................... 46
Fig 3. 8. Three RS-trees generated by the program for the text in example 3.5............. 47
Fig 4. 1. System architecture of the Arabic text summarizer based on RST............... 51
Fig 4. 2. The system packages.................................................................................... 53
Fig 4. 3. RST processes chain.................................................................................... 54
Fig 4. 4. Chain of RST processes pattern class structure.......................................... 55
Fig 4. 5. Multi-Relations hypothesizers pattern class structure.................................. 58
Fig 4. 6. Multi-RS-tree selectors pattern class structure............................................. 62
Fig 4. 7. Arabic text summarization system scenario.................................................. 65
Fig 4. 8. Arabic text summarizer data model............................................................. 66
Fig 4. 9. Arabic text summarization data tables......................................................... 67
Fig 5. 1. The behavior of the auto-summarization software........................................ 70
Chapter 1: Introduction

1 Motivation

Natural language processing became an essential field in computer science. The applications of natural language processing are getting high importance in the age of World Wide Web where people search for information, with the support of the evolution of the hardware specification. The need for new applications evolved from the emergence of the new technologies. Difference applications have been put into research focus including but not limited to information retrieval, text classification, text summarization, machine translation … etc.

Text summarization is one of the important applications of natural language processing which this thesis will be focusing on. It is common that when people read certain document or book, they want just to get the idea that this book or document is taking about. Businessmen don’t have enough time to read all the documents that include import information as well as other general information; with the help of an automatic text summarization system, they can get what are the idea behind those documents in a few mouse clicks. Researchers need such system when they scan the references to evaluate them in performing the task of annotated bibliography, and selecting the suitable references for their research. Normal users need such system in reading articles, news and any kind of text.

The above introduction shows the importance of such system for all kind of people of different interests and education levels. As a result, researchers start spending efforts to bring this application to the state of the art. Different challenges need to be resolved to bring such system in the real world. The efforts have been launched many years ago on the English language. Different techniques have been employed to come up with a reasonable system that can extract the most importance statements in the text. A few years ago, companies started producing automatic summarization systems and put them in the market, one of those systems is the summarizer that comes with Microsoft
word. Those system are not mature enough, and still don’t give what customer expects.
Therefore, research is still conducted extensively on this area. English language has the
most effort, the other languages, have also been considered recently. Unfortunately, there
is not clear effort has been spent on the Arabic language which hundreds of millions of
people are speaking.

The evolution of the Arabic applications and the increasing of the Arabic speaker
users in the past few years led to the importance of pushing this language in one the
research focuses of the automatic text summarization. This thesis is the answer to those
factors; it addresses the problem of automatic Arabic text summarization.

During the work on the area of the automatic summarization, different
techniques have been proposed, where some of them are based on statistical process of
the given text, and the others are based on linguistics process. Since Rhetorical Structure
Theory (RST) has shown encouraging results in text processing in general, and text
summarization specifically, this thesis will try to employee this theory as a base that the
Arabic text summarization will be built upon.

The aim of this thesis is to build an infrastructure of the Arabic text
summarization, so that new techniques and additional improvements could be built upon
this infrastructure. Another objective is to employee the rhetorical structure theory on the
Arabic language leading to giving the opportunity to use this theory in applications based
on the Arabic language processing. Finally, this thesis addresses implementing the
proposed techniques in a pattern-oriented manner to make the system flexible enough for
additional improvement and enhancement since this field is still a subject of research.

The major goal of this thesis is putting the Arabic language into the consideration
of the new research in the language processing, and encouraging the software houses to
develop applications based on Arabic text processing. The two approaches –theoretical,
and practical, were taken into consideration during the work to make the subject useful
for academic centers as well as the software houses.
2 Text Summarization Overview

People have different meanings of what the text summary is, but those meanings are relevant in away or another; people most likely agree that the summary should give the reader the main idea behind the text; however, different definitions of the summary have been given; the ANSC Z39 (1971) defined the summary as following:

- *Summary* is a restatement within a document (usually at the end) of its salient findings and conclusions, and is intended to complete the orientation of a reader who has studied the preceding text.

In (Sparck-Jones, 1999) another definition was given, this definition is based on the process of generating the summary rather than what it should contain, so the definition was the following:

- *Summary* is a reductive transformation of source text to summary text through content reduction by selection and/or generalization on what is important on the source.

After giving the definition, the process of text summarization was broken into three stages. (Fig. 1.1) shows the stages of the process of text summarization which are the following:

i. Interpretation of source text content to arrive at a source text representation.

ii. Transformation of source text representation into summary representation.

iii. Generation of summary from summary representation.

During the first stage, the source text is interpreted, first ‘locally’ at the level of individual sentences before being interpreted ‘globally’ to give the source text representation (Jones & Endres-Niggemeyer, 1995). During the second stage of the summarization process, the source text representation is transformed into the summary text representation. The final stage is generating the text summary from the summary representation.
The most explored of approaches in automatics summarization is that of sentences extraction or selection (Chuah, 2001). In (section 1.2.1) will discuss three techniques in sentences selection:

- Frequency/statistical technique.
- Lexical cohesion technique.
- Rhetorical structure theory technique.

Extracted sentences could be further edited to condense them; however, we will discuss the proposed condensation techniques in (section 1.2.2) even though condensation is beyond the scope of this thesis.

### 2.1 Sentences Selection Techniques

The first technique which is statistical based makes use of a mix of word occurrence, text cues and other features, without taking into consideration the meaning of the words (Chuah, 2001). It is based on the idea that text surface cues are the most obvious indication of text contents. The word frequency was the earliest approach used, but the importance of a word is not just its frequency in the document; therefore, the use of a simple word count to select the most important sentences to generate the summary is not satisfactory. As a refinement, mix of cue words (e.g. greatest, significant), indicator phrases (e.g. in this study, our report indicates that) and location in text have been used to enhance the sentences selection. However, statistical techniques don’t take into consideration the semantic continuity of the text; as a result, the summary is disjointed, even if the sentences themselves are complete. At the end, the algorithm whose formula
is able to pick out important sentences with the greatest probability produces the best selection (Chuah, 2001).

Lexical chain technique takes care of the way lexical units of a text cohere. Normally, lexical units cohere in two ways: (a) reiteration, and (b) collocation. The cohesive strength between lexical units could be computed based upon: (a) number of links between them, or (b) distance between them. When sentences with the greatest number of links, or from the strongest chain of related words are extracted, the extracted sentences represent the text contents. This technique falls between the full semantic interpretation and the frequency count (Chuah, 2001).

The third technique is the rhetorical structure theory (RST). As its name indicates this technique takes care of the rhetorical relations between the text spans. In (Mann & A. Thompson, 1988), a set of twenty three rhetorical relations have been identified. When the relations between the text spans are extracted, a full representation of the text is built in a binary tree structure, and this structure is called schema or RS-tree. A full explanation of this theory is given in (chapter 2).

When rhetorical structure theory was employed in text summarization, it has shown substantial improvements over the other techniques since it takes care of the semantic of the text through rhetorically analyzing the text spans; because of this, we decided to use it as a base for our proposed summarization approach.

2.2 Proposed Condensation Processes

When the text summarization is applied, the extracted summary could be condensed for further improvement, but that is beyond the current state of the art in automatic text summarization; however, some condensation processes have been proposed, and we will go over some of them. First of all, we need a proper definition of what condensation means, and the definition as it has was given in (Chuah, 2001) is the following:

- **Condense** to make something that is spoken or written shorter, by not giving as much detail or using fewer words to give the same information.
(Rush et al., 1971) shows that using punctuations as a guide, a coordinated or subordinated segment of a given sentence maybe truncated without affecting the rest of the sentence. Sentence $S$ is rendered more concise by truncating or deleting a segment of text. The result is a modified sentence $S'$ which maybe further truncated to $S$ (Tab. 1.1).

<table>
<thead>
<tr>
<th>Rule</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Paraphrase, truncation.</td>
<td>$S \rightarrow S'$: the house was beautiful in the winter, but it was more comfortable in the summer time. $\rightarrow$ [The house was beautiful in the winter]$_S' \rightarrow$ [The house was beautiful]$_S''$</td>
</tr>
<tr>
<td>2. Concatenation, embedding.</td>
<td>$S_1, S_2 \rightarrow S$</td>
</tr>
<tr>
<td>3. Fragmentation.</td>
<td>$S \rightarrow S_1, S_2$</td>
</tr>
</tbody>
</table>

Tab 1.1. Condensation rules identified in (Rush et al., 1971).

As it was mentioned in (Chuah, 2001) the earliest work that was carried out to improve the quality of the summarization generated from the extracted sentences techniques is that in (Mathis et al., 1973). The study proposed five rules to improve the readability of the generated summary (Tab. 1.2). Rule 1 and rule 2 combine sentences by coordination and by subordination respectively; rule 3 and rule 5 on graphical reference and context modification treat hanging reference to figures, tables and ordinal numbers; they replace the segment of text for which reference to a figure, table or linguistic unit cannot be found with an appropriate word. Rule 4 adds a remark in the summary indicating the number of references given in the document.

In (Maybury, 1995) abstraction and aggregation are suggested to be distinct type of condensation sub-processes. While abstraction replaces a series of events with a single event, aggregation factors out the units in common between them. While abstraction involves linguistics and/or world knowledge, aggregation requires knowledge of syntax. (Tab. 1.3) shows an example of this condensation process as it appears in (Chuah, 2001).
<table>
<thead>
<tr>
<th><strong>Rule</strong></th>
<th><strong>Example</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Combination of sentences by means of coordinate conjunction.</td>
<td>$S_1, S_2 \rightarrow S_1 \text{ coord } S_2$: The system exceeded the capacity of its present auxiliary equipment + The system was modified for further testing $\rightarrow$ The system exceeded the capacity of its present auxiliary equipment and was modified for further testing.</td>
</tr>
<tr>
<td>2. Combination of sentences by means of subordinate conjunction.</td>
<td>$S_1, S_2 \rightarrow S_1 \text{ subord } S_2$: A set of consecutive storage locations is called a memory block + A memory block is labeled by a single word called a codeword $\rightarrow$ A set of consecutive storage locations is called a memory block, which is labeled by a single word called a codeword.</td>
</tr>
<tr>
<td>3. Graphical reference rule</td>
<td>(a) Table 2 presents nine areas of endeavor and their associated disciplines $\rightarrow$ A table presents nine areas of endeavor and their associated disciplines.</td>
</tr>
<tr>
<td></td>
<td>(b) Figure 2 presents graphically the general model of information transfer $\rightarrow$ A figure presents graphically the general model of information transfer.</td>
</tr>
<tr>
<td>4. Reference tabulation</td>
<td>If N references are given in document, then generate “N references were given”, and if no references, then generate “No references were given”.</td>
</tr>
<tr>
<td>5. Context modification</td>
<td>(a) The second mechanism is structural change … $\rightarrow$ A mechanism is structural change…</td>
</tr>
<tr>
<td></td>
<td>(b) The second is that reactions to oxygen atoms … $\rightarrow$ Reaction to oxygen atoms …</td>
</tr>
</tbody>
</table>

Tab 1. 2. Condensation rules identified in (Mathis et al., 1973).

<table>
<thead>
<tr>
<th><strong>Condensation Process</strong></th>
<th><strong>Example</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Abstraction</td>
<td>Movement events + missiles firing + aborted mission $\rightarrow$ foiled attach event.</td>
</tr>
<tr>
<td>2. Aggregation.</td>
<td>Site A fired a missile at time $t$ + Site D fired a missile at time $t$ $\rightarrow$ Site A and site D were simultaneously fired at time $t$.</td>
</tr>
</tbody>
</table>

Tab 1. 3. Condensation process identified in (Maybury, 1995).
In (Jing & McKeown, 2000) six operation were identified to edit the extracted sentences. (Tab. 1.4) shows the identified operations as they were given in (Chuah, 2001).

<table>
<thead>
<tr>
<th>Operation</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Sentence reduction.</td>
<td>While it arrives, $X \rightarrow X$ (&quot;While it arrives&quot; was deleted).</td>
</tr>
<tr>
<td>2. Sentence combination.</td>
<td>$X + Y \rightarrow [X and Y]$</td>
</tr>
<tr>
<td>3. Syntactic transformation.</td>
<td>Subject in a sentence is moved from the end to the front.</td>
</tr>
<tr>
<td>4. Lexical paraphrasing</td>
<td>(a) Point out $\rightarrow$ Note.</td>
</tr>
<tr>
<td></td>
<td>(b) Fits squarely into $\rightarrow$ Hits the head on the nail.</td>
</tr>
<tr>
<td>4. (a) Generalization.</td>
<td>(a) A proposed new law that would require web publishers to obtain parental consent before collecting personal information from children $\rightarrow$ Legislation to protect children's privacy on-line.</td>
</tr>
<tr>
<td></td>
<td>(b) The white house's top drug official $\rightarrow$ Gen. Barry R. McCaffrev, the white house's top drug official.</td>
</tr>
<tr>
<td>6. Reordering.</td>
<td>Place an ending sentence in an article at the beginning of an abstract.</td>
</tr>
</tbody>
</table>

Tab 1.4. Rephrasing operations identified in (Jing & McKeown, 2000).

In (Saggion, 2000), a comparative study information in abstracts prepared by professionals, and information from particular structural sections of source document identified fifteen *transformations*, and they are given in (Tab. 1.5) as they appear in (Chuah, 2001).

<table>
<thead>
<tr>
<th>Transformation</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Syntactic verb transformation</td>
<td>(a) Finally we address $X \rightarrow$ Addresses X.</td>
</tr>
<tr>
<td></td>
<td>(b) In this paper we have presented $X \rightarrow$ Presents X.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>
| 2. Lexical verb transformation. | (a) It identifies $X \rightarrow X$ are discussed.  
(b) This article details $X \rightarrow X$ are described. |
| 3. Verb selection. | (a) Running $X \rightarrow$ The running of $X$ is described.  
(b) I define $X \rightarrow$ Gives an overview of $X$. |
| 4. Conceptual deletion. | (a) Section 2 gives $X \rightarrow$ Gives $X$.  
(b) In this paper we report $X \rightarrow$ Reports $X$. |
| 5. Concept re-expression | (a) We analyze $X \rightarrow$ Analyzes $X$.  
(b) Our genie system $X \rightarrow$ Genie $X$. |
| 6. Structural deletion. | Indeed, $X \rightarrow X$. |
| 7. Clause deletion. | (a) The work described in this paper addresses these by nothing that $X \rightarrow X$.  
(b) To emphasize this fact we say that $X \rightarrow X$. |
| 8. Parenthetical deletion. | (a) It will show how extending the designer's description of the information processing system (with a language that details how changes within the application occurs) can allow for the construction of applications that are self explanatory $\rightarrow$ Extending the designer's description of the information processing system can allow for the construction of applications that are self explanatory. |
| 9. Acronym expansion. | The work focuses on APIs $\rightarrow$ The word focuses on application programming interface. |
| 10. Abbreviation. | The future of digital imaging at the National Railway Museum $\rightarrow$ Discusses the future of digital imaging at NRM. |
| 11. Merge. | Protocol selection + Address mapping and connection management $\rightarrow$ Protocol selection, address mapping and connection management are also described. |
| 12. Split. | [This has resulted in a tesseral temporal reasoning system, based on tesseral addressing and using tesseral arithmetic, which offers the advantage that it is directly compatible with existing GIS technology] → [A tesseral temporal reasoning system has been designed, based on tesseral addressing and using tesseral arithmetic] + [It offers the advantage that is compatible with existing GIS technology] |
| 13. Complex reformulation. | [SCULPTOR – an intuitive 3D modeling tool] + [The motivation for our work is to invent a design environment for architects, based on the most recent hard-and software development] + [These are mainly virtual reality (VR) interaction tools, fast graphics libraries, and new approaches in artificial intelligence] → [SCULPTOR – an intuitive 3D modeling tool, is being developed to create a design environment for architects based on virtual interaction tools, fast graphics libraries, and new approaches in artificial intelligence]. |
(b) The university of Liverpool → The university of Liverpool, UK.  
(c) Maxcess library system, Inc. with Maxcess library system → Maxcess library system.  
(d) The first experiment → Experiment 1.  
(e) UK: regulation of cable TV → regulation of cable TV in the UK.  
(f) Integrating speech and natural language processing → The integration of speech and natural language processing.  
(g) The Austrian situation in the field of telecommunication infrastructure → The Austrian telecommunication infrastructure. |
| 15. No transformation. | - |

**Tab 1.5.** Rephrasing transformation identified in (Saggion, 2000)
3  Thesis Overview

After giving an introduction on the text summarization techniques, chapter two will be dedicated to the rhetorical structure theory which the Arabic text summarization technique that this thesis proposes is based on. We will explain this theory and how it is used to analyze texts in detail. Chapter three will explain the process of employing RST in Arabic language; and then how the result of this employment is used to build an infrastructure for Arabic text summarization. In chapter four, the proposed theories in chapter three will be put in the practical world. We will explain how the proposed theories in chapter three could be designed and implemented in a view of pattern-orientation, to facilitate future enhancement and improvement; moreover, the pattern-oriented development view will propose a general model for Arabic text summarization systems. Finally, chapter five will describe the evaluation of the Arabic text summarizer that has been implemented according to the theories explained in chapter three, and the design explained in chapter four; it will explain the evaluation process as well as the evaluation result. The diagram shown in (Fig 1.2) explains the process which this thesis followed.

![Fig 1.2. Thesis process.](image)
Chapter 2: Rhetorical Structure Theory

1 Introduction

A research had been conducted in Information Science Institute (ISI) at the University of Southern California (USC) to describe the written discourse. The aim of this research was finding a theoretical basis in designing computer program which is capable of generating texts. The effort was beyond this goal, and resulted in an understanding of discourse that has had many other uses, including several applications in linguistics. The effort involved a study of the nature of the text with an interesting in developing a theory of text structure that can serve both as an analytical tool as well as a text generation tool. The theory has been called Rhetorical Structure Theory (RST), since it provides a framework for describing rhetorical relations among parts of a text.

In the construction of RST, 400 texts were studied, and the observation was that many phenomena of text structure involved pairs of regions of the text. The mutual relevance of the two parts, and sometimes their position and form, could be identified with recurrent relations holding between the parts. These relations could hold between text parts of wide range of sizes, from clauses to groups of paragraphs. The observation resulted the following assumptions (Mann et al., 1992):

1. **Organization**: texts consist of functionally significant parts; the parts are elements of patterns in which parts are combined to create larger parts and whole text.

2. **Unity and coherence**: to be recognized as a text, the writing must create a sense of overall unity to which every part contributes.

3. **Unity and coherence arise from imputed function**: a text is perceived as having unity and coherence because all of its parts are seen as contributing to a single purpose of the writer.
4. **Hierarchy**: texts are organized such that elementary parts are composed into larger parts, which in turn are composed into yet larger parts up to the whole text.

5. **Homogeneity of hierarchy**: there is one set of structural patterns available for organizing the text at every scale, from the largest down to the smallest scale. This set of pattern is identified as *RST schema*.

6. **Relational composition**: the principle of structural pattern in multi-sentential text is relational: a small set of highly recurrent relations holding between pairs of parts is used to link parts together to form larger parts.

7. **Asymmetry of relations**: the most common type of text structuring relation is an asymmetric class, called *nucleus-satellite* relations in RST. This class is asymmetric because one member of pair of text spans is more central (the *nucleus*) and one more peripheral (the *satellite*).

8. **Nature of relations**: text structuring relations are functional; they can be described in term of the purpose of the writer, the writer assumptions about the reader, and certain propositional patterns in the subject matter of the text.

9. **The number of relations**: the set of text structuring relations is open, so that additional previously unused relations can arise.

1.1 **Rhetorical Relations**

The key elements of RST are relations and spans. In (Mann et al., 1992) the two terms were defined according to the RST as following:

- **Relation** –is the relationship that can hold between two text spans.

- **Text span** –is any portion the text that has an RST structure (and thus has a functional integrity, from a text-organizational point of view), or that is realized by a unit.

Relations are defined to hold between two non-overlapping text spans, and the relation definition consists of four fields (Mann & A.Thompson, 1988):
1. Constraints on the nucleus.
2. Constraints on the satellite.
3. Constraints on the combination of nucleus and satellite.
4. The effect.

Each field specifies particular judgments that the text analyst must make in building the RST structure. (Tab. 2.1) shows the definition of the evidence relations as it appears in (Mann et al., 1992). The effect field is classified into two classes depending on which spans take part in the desired effect (Mann & Matthiessen, 1991). The two classes are (a) *nucleus effect*-the desired effect involves the nucleus only, and (b) *nucleus and satellite effect*-the desired effect involves both the nucleus and the satellite. The term *locus of effect* refers to this difference; the locus of effect is either nucleus or the combination of nucleus and satellite.

<table>
<thead>
<tr>
<th>Constraints</th>
<th>Effects</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. <em>Constraints on the nucleus</em>: the reader might not believe the nucleus to a degree satisfactory to the writer.</td>
<td>a. The reader’s belief of the nucleus is increased.</td>
</tr>
<tr>
<td>b. <em>Constraints on the satellite</em>: the reader believes the satellite or will find it credible.</td>
<td>b. <em>Locus of the effect</em>: Nucleus.</td>
</tr>
<tr>
<td>c. <em>Constraints on the combination of nucleus and satellite</em>: the reader’s comprehending the satellite increases his or her belief of the nucleus.</td>
<td></td>
</tr>
</tbody>
</table>

**Tab 2.1.** The definition of the evidence relation in (Mann et al., 1992).

Based upon a study on the English language, a set of twenty one rhetorical relations have been identified in (Mann & Matthiessen, 1991); those relations are stated in (Tab. 2.2). The list of relations is open for further expansion based on the ninth assumption mentioned above.
<table>
<thead>
<tr>
<th>Relations name</th>
<th>Locus of effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elaboration</td>
<td>Nucleus + Satellite</td>
</tr>
<tr>
<td>Circumstance</td>
<td>Nucleus + Satellite</td>
</tr>
<tr>
<td>Solutionhood</td>
<td>Nucleus + Satellite</td>
</tr>
<tr>
<td>Volitional cause</td>
<td>Nucleus + Satellite</td>
</tr>
<tr>
<td>Volitional result</td>
<td>Nucleus + Satellite</td>
</tr>
<tr>
<td>Non-volitional cause</td>
<td>Nucleus + Satellite</td>
</tr>
<tr>
<td>Non-volitional result</td>
<td>Nucleus + Satellite</td>
</tr>
<tr>
<td>Purpose</td>
<td>Nucleus + Satellite</td>
</tr>
<tr>
<td>Condition</td>
<td>Nucleus + Satellite</td>
</tr>
<tr>
<td>Otherwise</td>
<td>Nucleus + Satellite</td>
</tr>
<tr>
<td>Interpretation</td>
<td>Nucleus + Satellite</td>
</tr>
<tr>
<td>Evaluation</td>
<td>Nucleus + Satellite</td>
</tr>
<tr>
<td>Restatement</td>
<td>Nucleus + Satellite</td>
</tr>
<tr>
<td>Summary</td>
<td>Nucleus + Satellite</td>
</tr>
<tr>
<td>Evidence</td>
<td>Nucleus</td>
</tr>
<tr>
<td>Antithesis</td>
<td>Nucleus</td>
</tr>
<tr>
<td>Concession</td>
<td>Nucleus</td>
</tr>
<tr>
<td>Motivation</td>
<td>Nucleus</td>
</tr>
<tr>
<td>Enablement</td>
<td>Nucleus</td>
</tr>
<tr>
<td>Justify</td>
<td>Nucleus</td>
</tr>
<tr>
<td>Background</td>
<td>Nucleus</td>
</tr>
</tbody>
</table>

**Tab 2.2.** English rhetorical relations as stated in (Mann & Matthiessen, 1991).
1.2 RST Schema

Schemas define the structural constituency arrangement of text (Mann & A. Thompson, 1988). They are abstract patterns consisting of a small number of a constituent text spans, a specification of the relations between them, and a specification of how certain spans (nuclei) are related to the who collection (Marcu, 1997c, Marcu, 1997d, Marcu, 2000a). (Fig. 2.1) shows an example of five schemas given in (Mann & A. Thompson, 1988). The curves in (Fig. 2.1) represent relations holding, and the straight lines identify the nuclear spans. There are three conventions that determine the possible applications of a schema.

1. **Unordered spans**: the schemas do not constrain the order of nucleus or satellite in the text span in which the schema is applied.

2. **Optional relations**: for multi-relation schemas, all individual relations are optional, but at least one of the relations must hold.

3. **Repeated relations**: a relation that is part of a schema can be applied any number of times in the application of that schema.

![Fig 2.1. Example of five schemas.](image)
1.3 Nuclearity

The discussion of nuclearity so far simply uses nucleus and satellite as span labels in definitions. But they represent a much more pervasive and significant regularity. There is a characteristic difference across relations in how the two spans function (Mann & Thompson, 1987):

a. One span is more prominent than the other.
b. One span is more essential to the text than the other.
c. For schemas with multiple relations, there is a single core span that all of the other spans are related to.
d. The identity of the prominent and essential span can be predicated from the relation itself. It is not conditional on the content of the spans or their context.

The prominent and essential core span is called the nucleus, and the other remaining spans the satellite. One way to recognize the functional distinctiveness of nuclei and satellites is to examine the effects of perturbing texts (Mann & A.Thompson, 1988):

1. **Nucleus deletion and nuclear function**: we predict that if a particular nucleus is removed, then the significance of the material in its satellite(s) will not be apparent.

2. **Satellite deletion and nuclear function**: if units that only function as satellites and never as nuclei are deleted, we should still have a coherent text with a message resembling that of that of the original; it should be something like a synopsis of the original text.

As discussed so far the most common type of relations those that connect nucleus span with satellite span, and those are called *hypotactic relations* (e.g. the relations evidence). But still there are some relations that are multi-nuclear, which connect two nucleus spans, and they are called *paratactic relations* (e.g. the relation joint) (Marcu, 1997e, Marcu, 2000b).
1.4 RST Analysis

RST analysis is a structural analysis of a text which is a set of schema applications such that the following constraints hold (Mann & A. Thompson, 1988):

- **Completeness**: the set contains one schema application that contains a set of text spans that constitute the entire text.
- **Connectedness**: except for the entire text as a text span, each text span in the analysis is either a minimal unit or constituent of another schema application of the analysis.
- **Uniqueness**: each schema application consists of a different set of text spans, and within multi-relations schema, each relation applies to different set of text spans.
- **Adjacency**: the text spans of each schema application constitute one text span.

The completeness, connectedness and uniqueness taken together are sufficient to cause RST analysis to be trees.

The first step in analyzing a text is to divide it into *units*. Unit size is arbitrary in RST. The next step is to identify spans and relations, working either the top down (progressive refinement) or from the bottom up (aggregation). It often happens that text has more than one analysis; it is normal and predictable outcome, given the way that RST is defined. There are several different causes of this multiplicity (Mann et al., 1992):

1. Boundary judgments: results of forcing borderline cases into categories.
2. Text structure ambiguity: comparable to many other varieties of linguistic ambiguity.
3. Simultaneous analysis: multiple compatible analyses.
4. Differences between analysts.
5. Analytical error.

In (Mann & A. Thompson, 1988) a wide range of text were analyzed, and the conclusion was the following:
• Virtually every text has an RST analysis.

• Certain text types characteristically do not have RST analyses. These include laws, contracts, reports ‘for the record’ and various kind of language-as-art, including poetry.

• Texts that have RST analyses predominate. It is thus typical, but not universal, for texts to be hierarchically structured and functionally organized.

An example of the RST analysis is shown below, this example was taken from (Marcu, 2000b). The text is divided into units (the units are numbered in the example), and then the relations are identified; the result is the RST schema shown in (Fig. 2.2).

**Example 2.1:**

[There are two houses you might be interested in:]\(^1\)

[House A is in Palo Alto.]\(^2\)[It's got 3 bedrooms and 2 paths, ]\(^3\)[and was built in 1950.]\(^4\)[It's on a quarter acre, with a lovely garden, ]\(^5\)[and the owner is asking $425K.]\(^6\)[But that's all I know about it.]\(^7\)

[House B is in Portola Valley.]\(^8\)[It's got 3 bedrooms, 4 baths and a kidney-shape pool, ]\(^9\)[and was also built in 1950]\(^10\)[It's on 4 acres of steep wooded slope, with a view of the mountain.]\(^11\)[The owner is asking $600K.]\(^12\)[I heard all this from a friend, ]\(^13\)[who saw the house yesterday.]\(^14\)[Is that enough information for you to decide which to look at?]\(^15\)

**Fig 2.2.** RST analysis for the example 2.1.
2 Building the Rhetorical Structure Tree (RS-tree)

When the text units are identified, and the rhetorical relations that hold between those text units are recognized, the RS-tree could be built accordingly. In (Marcu, 2000b) complete formalization of RS-tree is given which will be described in (section 2.1); (section 2.2) will demonstrate the process of building the RS-tree.

2.1 RS-tree formalization

The formalization uses the following predicates:

- Predicate *position* \((u_i, j)\) is true for textual unit \(u_i\) in sequence \(U\) if and only if \(u_i\) is the \(j^{th}\) element in the sequence.

- Predicate *rhet_rel* \((name, u_i, u_j)\) is true for textual unit \(u_i\) and \(u_j\) with respect to rhetorical relation *name* if and only if the definition \(D\) of rhetorical relation *name* is consistent with the relation between textual unit \(u_i\), in most cases satellite, and \(u_j\), a nucleus.

- Predicate *rhet_rel_ext* \((name, s_s, s_e, n_s, n_e)\) is true for textual spans \([s_s, s_e]\) and \([n_s, n_e]\) with respect to rhetorical relations *name* if and only if the definition \(D\) of rhetorical relation *name* is consistent with the relations between the textual spans that ranges over units \(s_s\)-\(s_e\), in most cases a satellite, and units \(n_s\)-\(n_e\), a nucleus.

The formalization proposed is built on the following features:

- A text tree is a binary whose leaves denote elementary textual units.

- Each node has associated a *status* (nucleus or satellite), a *type* (the rhetorical relation that holds between the text spans that that node spans over), and a *salience* or *promotion set* (the set of units that constitute the most important part of the text that is spanned by the node). (Fig. 2.3) shows and example of these features.
Fig 2. 3. Example of the feature status, type, and promotion for an evidence relation that connects two leaf spans.

According to (Marcu, 2000b), there are some constraints that hold in building the RS-tree, and those constraints are:

- For every span \([l, h]\), the set of objects over which the status ranges is the set \(nucleus, satellite, \) and none.
- The status of any text span is unique.
- For every span \([l, h]\), the set of objects over which type ranges is the set of rhetorical relations that are relevant to that span.
- At most one rhetorical relation can connect two adjacent spans.
- For every span \([l, h]\), the set of objects over which promotion ranges is the set of units that make up that span.
- Text spans do not overlap.
- A text span with status none doesn't participate in the tree at all.
- There exists a text span, the root, which spans over the entire text.
- The status, the type, and the promotion set that are associated with a text span reflect the compositionality criterion.
2.2 Building the RS-tree

A proof-theoretic account for the valid text structure is given in (Marcu, 1997a, Marcu, 2000b), and it is used to derive all the valid RS-trees by a given algorithm. Marcu presented axioms and then an algorithm that uses the given axioms to derive all the valid RS-trees. The axioms use the following form for the textual spans:

\[
S(l, b, \text{tree} \ (\text{status}, \text{type}, \text{promotion}, \text{left}, \text{right}), \text{rr})
\]

Where \([l, b]\) is the span boundary, and the predicate \text{tree} is the valid text structure for that span, and \text{rr} is the set of rhetorical relations which can be used to get one relation that connects this span with another one. We express the axioms in pseudo-code as following [Note: we will not use the operators "exclusively belongs to" and "exclusive relation subtraction" defined and used in by Marcu; we will use the normal "belongs to" and "relations subtraction" instead]:

- If \([S_1(l, b, \text{tree}_1(Satellite, type_1, p_1, left_1, right_1), \text{rr}_1), and \]
  \(S_2(b+1, h, \text{tree}_2(Nucleus, type_2, p_2, left_2, right_1), \text{rr}_2), and \]
  \(rhet\_rel(name, s, n) \text{ belongs to } \text{rr}_1, and \]
  \(rhet\_rel(name, s, n) \text{ belongs to } \text{rr}_2, and \]
  \(s \text{ belongs to } p_1, and \ n \text{ belongs to } p_2, and \]
  \(\text{hypotactic(name)\}]
  \rightarrow \{S_3(l, h, \text{tree}(Nucleus, name, p_2, tree_1, tree_2), (\text{rr}_1 \cap \text{rr}_2) - \{rhet\_rel(name, s, n)\})\}
\]

axiom (1)

- If \([S_1(l, b, \text{tree}_1(Satellite, type_1, p_1, left_1, right_1), \text{rr}_1), and \]
  \(S_2(b+1, h, \text{tree}_2(Nucleus, type_2, p_2, left_2, right1), \text{rr}_2), and \]
  \(rhet\_rel(name, s, n) \text{ belongs to } \text{rr}_1, and \]
  \(rhet\_rel(name, s, n) \text{ belongs to } \text{rr}_2, and \]
  \(s \text{ belongs to } p_1, and \ n \text{ belongs to } p_2, and \]
  \(\text{hypotactic(name)\}]
\]

axiom (2)
→ \{S_3(l, h, \text{tree}(\text{Satellite, name, } p_2, \text{tree}_1, \text{tree}_2)), (rr_1 \cap rr_2) - \{\text{rhet\_rel(name, s, n)}\}\} \tag{axiom (2)}

- If \{S_1(l, b, \text{tree}_1(\text{Satellite, type}_1, p_1, \text{left}_1, \text{right}_1), \text{rr}_1), and \\\n\text{S}_2(b+1, h, \text{tree}_2(\text{Nucleus, type}_2, p_2, \text{left}_2, \text{right}_1), \text{rr}_2), and \\\nrhet\_rel\_ext(name, l, b, b+1, h) \text{ belongs to } \text{rr}_1, \text{ and} \\\nrhet\_rel\_ext(name, l, b, b+1, h) \text{ belongs to } \text{rr}_2, \text{ and} \\\nhypotactic(name)\} \\\n→ \{S_3(l, h, \text{tree}(\text{Nucleus, name, } p_2, \text{tree}_1, \text{tree}_2)), (rr_1 \cap rr_2) - \{\text{rhet\_rel(name, l, b, b+1, h)}\}\} \tag{axiom (3)}

- If \{S_1(l, b, \text{tree}_1(\text{Satellite, type}_1, p_1, \text{left}_1, \text{right}_1), \text{rr}_1), and \\\n\text{S}_2(b+1, h, \text{tree}_2(\text{Nucleus, type}_2, p_2, \text{left}_2, \text{right}_1), \text{rr}_2), and \\\nrhet\_rel\_ext(name, l, b, b+1, h) \text{ belongs to } \text{rr}_1, \text{ and} \\\nrhet\_rel\_ext(name, l, b, b+1, h) \text{ belongs to } \text{rr}_2, \text{ and} \\\nhypotactic(name)\} \\\n→ \{S_3(l, h, \text{tree}(\text{Nucleus, name, } p_2, \text{tree}_1, \text{tree}_2)), (rr_1 \cap rr_2) - \{\text{rhet\_rel(name, l, b, b+1, h)}\}\} \tag{axiom (4)}

- If \{S_1(l, b, \text{tree}_1(\text{Nucleus, type}_1, p_1, \text{left}_1, \text{right}_1), \text{rr}_1), and \\\n\text{S}_2(b+1, h, \text{tree}_2(\text{Satellite, type}_2, p_2, \text{left}_2, \text{right}_1), \text{rr}_2), and \\\nrhet\_rel(name, s, n) \text{ belongs to } \text{rr}_1, \text{ and} \\\nrhet\_rel(name, s, n) \text{ belongs to } \text{rr}_2, \text{ and} \\\ns \text{ belongs to } p_2, \text{ and } n \text{ belongs to } p_1, \text{ and} \\\nhypotactic(name)\} \\\n→ \{S_3(l, h, \text{tree}(\text{Nucleus, name, } p_1, \text{tree}_1, \text{tree}_2)), (rr_1 \cap rr_2) - \{\text{rhet\_rel(name, s, n)}\}\} \tag{axiom (5)}

- If \{S_1(l, b, \text{tree}_1(\text{Nucleus, type}_1, p_1, \text{left}_1, \text{right}_1), \text{rr}_1), and
\[ S_2(b+1, h, \text{tree}_2(\text{Satellite}, \text{type}_2, p_2, \text{left}_2, \text{right}_1), \text{rr}_2), \text{and} \]
\[ \text{rhet\_rel}(\text{name}, s, n) \text{ belongs to } \text{rr}_1, \text{and} \]
\[ \text{rhet\_rel}(\text{name}, s, n) \text{ belongs to } \text{rr}_2, \text{and} \]
\[ s \text{ belongs to } p_2, \text{and } n \text{ belongs to } p_1, \text{and} \]
\[ \text{hypotactic}(\text{name})] \]
\[ \rightarrow [S_3(l, h, \text{tree}(\text{Satellite}, \text{name}, p_1, \text{tree}_1, \text{tree}_2), (\text{rr}_1 \cap \text{rr}_2) \setminus \{\text{rhet\_rel}(\text{name}, s, n)\})] \text{ axiom (6)} \]

- \[ \text{If } [S_1(l, b, \text{tree}_1(\text{Nucleus}, \text{type}_1, p_1, \text{left}_1, \text{right}_1), \text{rr}_1), \text{and} \]
\[ S_2(b+1, h, \text{tree}_2(\text{Satellite}, \text{type}_2, p_2, \text{left}_2, \text{right}_1), \text{rr}_2), \text{and} \]
\[ \text{rhet\_rel\_ext}(\text{name}, b+1, h, l, b) \text{ belongs to } \text{rr}_1, \text{and} \]
\[ \text{rhet\_rel\_ext}(\text{name}, b+1, h, l, b) \text{ belongs to } \text{rr}_2, \text{and} \]
\[ \text{hypotactic}(\text{name})] \]
\[ \rightarrow [S_3(l, h, \text{tree}(\text{Satellite}, \text{name}, p_1, \text{tree}_1, \text{tree}_2), (\text{rr}_1 \cap \text{rr}_2) \setminus \{\text{rhet\_rel}(\text{name}, \text{b+1}, h, l, b)\})] \text{ axiom (7)} \]

- \[ \text{If } [S_1(l, b, \text{tree}_1(\text{Nucleus}, \text{type}_1, p_1, \text{left}_1, \text{right}_1), \text{rr}_1), \text{and} \]
\[ S_2(b+1, h, \text{tree}_2(\text{Satellite}, \text{type}_2, p_2, \text{left}_2, \text{right}_1), \text{rr}_2), \text{and} \]
\[ \text{rhet\_rel\_ext}(\text{name}, b+1, h, l, b) \text{ belongs to } \text{rr}_1, \text{and} \]
\[ \text{rhet\_rel\_ext}(\text{name}, b+1, h, l, b) \text{ belongs to } \text{rr}_2, \text{and} \]
\[ \text{hypotactic}(\text{name})] \]
\[ \rightarrow [S_3(l, h, \text{tree}(\text{Satellite}, \text{name}, p_1, \text{tree}_1, \text{tree}_2), (\text{rr}_1 \cap \text{rr}_2) \setminus \{\text{rhet\_rel}(\text{name}, \text{b+1}, h, l, b)\})] \text{ axiom (8)} \]

- \[ \text{If } [S_1(l, b, \text{tree}_1(\text{Nucleus}, \text{type}_1, p_1, \text{left}_1, \text{right}_1), \text{rr}_1), \text{and} \]
\[ S_2(b+1, h, \text{tree}_2(\text{Satellite}, \text{type}_2, p_2, \text{left}_2, \text{right}_1), \text{rr}_2), \text{and} \]
\[ \text{rhet\_rel}(\text{name}, n_1, n_2) \text{ belongs to } \text{rr}_1, \text{and} \]
\[ \text{rhet\_rel}(\text{name}, n_1, n_2) \text{ belongs to } \text{rr}_2, \text{and} \]
n₁ belongs to p₁, and n belongs to p₂, and

paratactic(name)

→ \{S₃(l, h, tree(Nucleus, name, p₁ ∪ p₂, tree₁, tree₂), (rr₁ ∩ rr₂) – \{rhet_rel(name, s, n)\})\}

axiom (9)

- If \(S₁(l, b, tree₁(Nucleus, type₁, p₁, left₁, right₁), rr₁)\), and \(S₂(b+1, h, tree₂(Nucleus, type₂, p₂, left₂, right₁), rr₂)\), and

rhet_rel(name, n₁, n₂) belongs to rr₁, and

rhet_rel(name, n₁, n₂) belongs to rr₂, and

n₁ belongs to p₁, and n belongs to p₂, and

paratactic(name)

→ \{S₃(l, h, tree(Satellite, name, p₁ ∪ p₂, tree₁, tree₂), (rr₁ ∩ rr₂) – \{rhet_rel(name, s, n)\})\}

axiom (10)

- If \(S₁(l, b, tree₁(Nucleus, type₁, p₁, left₁, right₁), rr₁)\), and \(S₂(b+1, h, tree₂(Nucleus, type₂, p₂, left₂, right₁), rr₂)\), and

rhet_rel_ext(name, l, b, b+1, h) belongs to rr₁, and

rhet_rel_ext(name, l, b, b+1, h) belongs to rr₂, and

paratactic(name)

→ \{S₃(l, h, tree(Nucleus, name, p₁ ∪ p₂, tree₁, tree₂), (rr₁ ∩ rr₂) – \{rhet_rel(name, s, n)\})\}

axiom (11)

- If \(S₁(l, b, tree₁(Nucleus, type₁, p₁, left₁, right₁), rr₁)\), and \(S₂(b+1, h, tree₂(Nucleus, type₂, p₂, left₂, right₁), rr₂)\), and

rhet_rel_ext(name, l, b, b+1, h) belongs to rr₁, and

rhet_rel_ext(name, l, b, b+1, h) belongs to rr₂, and

paratactic(name)
The axioms are used by the algorithm shown in (Fig. 2.4) to derive all the valid RS-trees. This algorithm was given in (Marcu, 2000b), but we changed this algorithm according to the tuning we made to the axioms.

**Input**: a text \( T \) of \( N \) units and set \( RR \) of rhetorical relations that hold among these units.

**Output**: all the RS-trees for the input text.

for \( i = 1 \) to \( N \)
Write each unit as:
- \( S(i, i, \text{tree(Satellite, Leaf, \{u_i\}, Null, Null, RR}) \)
- \( S(i, i, \text{tree(Nucleus, Leaf, \{u_i\}, Null, Null, RR}) \)

for size_of_span= 1 to N-1
 for \( l = 1 \) to \( N \cdot \text{size_of_span} \)
 h= \( l + \text{size_of_span} \)
 for \( b=1 \) to \( h-1 \)
 for each \( S(l, b, \text{tree}_1, RR_1) \) of span [1, b]
 for each \( S(b+1, h, \text{tree}_2, RR_2) \) of span [b+1, h]
 for each relation \( r \) such that \( r \) belongs to \( RR_1 \) and \( r \) belongs to \( RR_2 \)
 apply all possible axioms (1)-(12)

**Fig 2.4.** Algorithm that derive all the RS-trees for a given text.

### 3 RS-tree and Text Summarization

Researchers in computational linguistics have long speculated that the nuclei of a RS-tree constitute an adequate summarization of the text for which that tree was built (Marcu, 1999); the elementary units in the promotion set of a node of a tree structure which depends on the nuclear statuses of its immediate children, denote the most important units of the textual span that is dominated by that node (Marcu, 1997b, Marcu, 1998, Marcu, 1999). Therefore, the units in the promotion set of the root node are considered to be the most important units in the text; thus, the units in the promotion set
of a node in a certain level are considered to be more important than units in the promotion set of nodes in lower levels. This enables the construction of text summaries with various granularities (Marcu, 1998). For example, if we want a very short summary, we can create a summary text with units in the promotion set of the root node; if we want a longer summary, we can create a summary text with the units in the promotion set of the root node and the nodes in the first level of the RS-tree and so on.
Chapter 3: Arabic Text Summarization Theory

In this chapter we emphasize on the Arabic text summarization theory. It includes the Arabic rhetorical relations, the parsing of the Arabic text, and the process of the Arabic text summarization.

Each language has its own specifications; a theory in one language might not work with another if it is taken as it is. The same thing is applied to the rhetorical structure theory if it is applied to the Arabic language. Since this is the first attempt to apply RST on Arabic language, we will start by the foundation of the Arabic rhetorical relations, which (section 1) will be taking about. Once the Arabic rhetorical relations are found, the Arabic text can be rhetorically parsed and analyzed; (section 2) will be dedicated for this topic. Finally, when the process of analyzing the Arabic text is identified, the process of summarizing the Arabic text can be built upon, which is discussed in (section 3).

1 Arabic Rhetorical Relations

In (chapter 2), we demonstrated the foundation of the RST; during the process of the RST foundation, a set of rhetorical relations have been identified. The process of identifying the rhetorical relations includes corpus analysis. Since the analysis was done on the English corpus, the rhetorical relations that were identified can serve in English text analysis. But there is no guarantee that the same set of relations work on other languages. Since our aim is to work on the Arabic language, we have studied those relations in the Arabic corpus and Arabic literature. The result that we come up with is that the English rhetorical relations can not be taken as they are to serve in the rhetorical parsing of the Arabic text. For example, the two relations "concession" is not known in the Arabic rhetoric and literature. This relation and the relation "contrast" might correspond to the Arabic relation "امتداد"; the following examples explain the idea:
**Example 3.1:**

The following text was taken from (Mann et al., 1992) as an example of the relation concession:

"[Concern that this material is harmful to health or the environment may be misplaced.]¹
[Although it is toxic to certain animals, evidence is lacking that it has any serious long-term effect on human beings.]²"

**Translation to Arabic:**

[اعتبر هذه المادة ضارة بالصحة أو أن البيئة قد تكون بها.]¹ [مع أنها سامة لبعض الحيوانات، لا يوجد دليل على أن لها أي تأثيرات طويلة المدى على الإنسان.]²

**Example 3.2:**

The following text was taken from (Mann et al., 1992) as an example of the relation contrast (between units 1 and 2):

"[Animals heal.]¹ [but trees compartimentalize.]² [They endure a lifetime of injury and infection by setting boundaries that resist the spread of the invading microorganisms.]³"

**Translation to Arabic:**

[الحيوانات تتعافى.]¹ [لكن الأشجار ينقسمون.]² [فهُم يتحملون الجروح و الإصابات طوال حياتهم بوضع حدود تقاوم انتشار الجراثيم.]³

Because of the difference in the rhetoric, literature, and relation concepts between the two languages, we decided to study the Arabic corpus to extract some Arabic rhetorical relations that we can work on. The approaches we used to extract the relation is shown in (section 1.1).

### 1.1 Rhetorical Relations Foundation

Three approaches we have followed to extract the Arabic rhetorical relations; these approaches are:

- From the English relations.
- From the Arabic corpus.
- From the Arabic cues.

First, we extracted some of the Arabic relations from the English relations. The process consists of three steps shown in (Fig. 3.1). We pick an English relation, then scan the Arabic rhetoric and literature references (Abdulmuttalib, 2003, Aubadah, 1983, Gabawah, 1972) for this relation, we also scan the Arabic corpus to see if this relation is explicitly signaled; if so, the relation is added to the Arabic relations list; otherwise, the relations is ignored.

![Flowchart](image)

**Fig 3.1.** The process of extracting the Arabic relations from English relations.
The second approach is that we search the Arabic rhetoric and literature references that have been written by Arabic language scholar for the relations that connect the Arabic clauses. Those relations fall into two categories:

- Connectors that connect clauses as well as words.
- Connectors that connect clauses only.

The following examples explain the two categories:

**Example 3.3:**
The following sentence includes the relation "جار و مجرور" that connects a clause with a word.

وَجِدَتْ يَزِيدُ فِي السَّيْرَةِ.

The following sentence includes the same relation, but in this case it connects two clauses.

وَجِدَتْ يَزِيدُ فِي بَيْتٍ أَخِيهِ.

**Example 3.4:**
The following sentence includes a relation "استدراك" that connects two clauses only.

سُءْدَّى البَيْتِ إِلَى الْمَكَّةِ، وَلَكِنِّي لَنْ أَحْضَرِ الْإِضْرَابَ.

We select the relations that belong to the second category since we are targeting the relations between the clauses or sentences.

In the third approach, we scan the Arabic connector words (cues) (Al-Ansari, 2003), and then examine the relations that they signal; we check those relations if they belong to the second category, then we add them to the Arabic relations; otherwise we ignore them. The examples (3.3 and 3.4) explain this approach; the connectors (underlined) are extracted, then the relations they signal are studied and then we make the decision of including these relations or not.

At the end of going through the three approaches we ended up with eleven relations shown in (Tab. 3.1).
<table>
<thead>
<tr>
<th>Arabic Rhetorical Relations</th>
<th>Name in English</th>
<th>Type</th>
<th>Explanation</th>
</tr>
</thead>
</table>
| شرط                         | Condition      | Hypotactic | **Constraints on S:** S presents a condition whose realization would lead to the realization of N.  
**Constraints on N:** N is a situation which will be realized in case of the realization of S.  
*Example:* سأشترى ثلاثة كتب إذا ذهب إلى المكتبة.

| تفسير                      | Interpretation | Hypotactic | **Constraints on S:** S presents an interpretation for a situation presented in N which the writer thinks that it is not clear to the reader.  
**Constraints on N:** N is a situation which the writer wants to convey to the reader and thinks that it is not clear.  
*Example:* سافر محمد إلى مدينة جدة في الساعة الخامسة، أي أنه سيصل في الساعة السابعة تقريبا.

| تعليل                       | Justification  | Hypotactic | **Constraints on S:** S presents a justification that the writer thinks that reader might not believe the situation in N without it.  
**Constraints on N:** N is a situation which the reader needs a proof to believe it.  
*Example:* تم تأجيل المباراة بسبب هطول الأمطار.
<table>
<thead>
<tr>
<th>Arabic</th>
<th>English</th>
<th>Type</th>
<th>Hypotactic Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>استدراك</td>
<td>Recalling</td>
<td>Constraints on S: S presents an annotation which the writer wants to mention in conjunction with the situation in N. Constraints on N: none. Example: هذه السيارة جميلة ولكنها غالبة الثمن.</td>
<td></td>
</tr>
<tr>
<td>توكيد</td>
<td>Confirmation</td>
<td>Constraints on S: S presents a confirmation to a situation in N. Constraints on N: N is a situation which the writer thinks that the reader might doubt it. Example: سأسترئي ثلاثة كتب إذا ذهبت إلى المكتبة.</td>
<td></td>
</tr>
<tr>
<td>نتيجة</td>
<td>Result</td>
<td>Constraints on S: S presents some parameters which leads to a certain fact. Constraints on N: N is a fact that is achieved by the achievement of the parameters in S. Example: لم يتم وضع الخطة المناسبة في المباراة، و نتيجة لذلك خسر الفريق المباراة بنتيجة كبيرة.</td>
<td></td>
</tr>
<tr>
<td>تمثيل</td>
<td>Example</td>
<td>Constraints on S: S presents an example for a situation in N. Constraints on N: none. Example: بعض الألوان تستخدم في التنبيه مثل اللون الأصفر والأحمر.</td>
<td></td>
</tr>
<tr>
<td>Arabic Rhetorical Relations</td>
<td>Base</td>
<td>Hypotactic</td>
<td></td>
</tr>
<tr>
<td>-----------------------------</td>
<td>------</td>
<td>------------</td>
<td></td>
</tr>
<tr>
<td><strong>Constraints on S:</strong> S presents a base that the situation in N will be built upon.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Constraints on N:</strong> N presents a situation that is built upon the situations in S.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong> بما أنك لا تملك سيارة، فسأقوم بإيصالك إلى البيت.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Arabic Rhetorical Relations</th>
<th>Explanation</th>
<th>Hypotactic</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Constraints on S:</strong> S presents an explanation for a situation presented in N.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Constraints on N:</strong> none.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong> لا أفضل الخروج من المنزل، لاسيما وأن الجو حار.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Arabic Rhetorical Relations</th>
<th>Joint</th>
<th>Paratactic</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Constraints on N:</strong> N presents multi-nuclear that writer thinks that they are important to the reader.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong> هذا الكتاب مفيد. ويمكن شراؤه من جميع المكتبات.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Arabic Rhetorical Relations</th>
<th>Sequence</th>
<th>Paratactic</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Constraints on N:</strong> N presents multi-nuclear that come in sequence depending on time or place.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong> سأذهب إلى العمل، ثم سأذهب لأشتري احتياجات المنزل.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Tab 3.1.** Arabic rhetorical relations.

### 1.2 Relations Signaling

Each relation is signaled by a set of cue phrases (Marcu, 1997e, Marcu, 2000b). We studied the Arabic rhetorical relations shown in (Tab 3.1) and observed the cue phrases that signal each one. We generated a set of cue phrases shown in (Appendix I) and employed them in the automatic Arabic text summarizer program. Each cue phrase signals a rhetorical relation between two units based on some features. The values of the
features are extracted from the corpus analysis of the cue phrases. Following are the cue phrases features:

- **Status**: specifies the rhetorical status of the units that are linked by the cue phrase. Its value must be either `satellite_nucleus`, or `nucleus_satellite` indicating the either the designated cue phrase connect two units where the first one is satellite and the second one is nucleus (if the value is `satellite_nucleus`) or the first one is nucleus and the second one is satellite (if the value is `nucleus_satellite`).

- **Position**: specifies the position in the text where the cue phrase must be located. Its value is either `beginning` indicating that the designated cue phrase is located in the beginning of the statement, or `middle` indicating that the cue phrase is located in the middle of the statement.

- **Action**: specifies the action that this cue phrase has in determining the elementary units. We discuss the values of this feature in (section 2.1).

- **Relation**: specifies the relation that this cue phrase signals.

- **Regular expression**: specifies the regular expression of the cue phrase.

(Tab 3.2) shows an example of the described features for the cue phrase "إن". As shown, when the cue phrase comes in the beginning of the statement, it connects the unit that the cue phrase "إن" is located in as satellite with the one that comes next to it as nucleus. The other case is when the cue phrase "إن" is located in the middle of the statement, it connect the unit the comes before the unit that this cue phrase is located in as nucleus, with the unit where the cue is located as satellite [Note: the regular expression is written using the java regular expression].
<table>
<thead>
<tr>
<th>Regular Expression</th>
<th>Position</th>
<th>Relation</th>
<th>Status</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>إن (ان)</td>
<td>B</td>
<td>شرط</td>
<td>S_N</td>
<td>Normal then Comma</td>
</tr>
<tr>
<td>كلاملا (كلام،)</td>
<td>M</td>
<td>شرط</td>
<td>N_S</td>
<td>Normal</td>
</tr>
</tbody>
</table>

Tab 3.2: Example of the features of the cue phrases

2 Rhetorical Parsing of Arabic Text

2.1 Elementary Units Determination

Before building the rhetorical relations, the text units that those relations work on should be determined. Since the rhetorical relations are signaled by specific cue phrases, the determination of the text units is based on cue phrases too (Marcu, 1997e, Marcu, 2000b). The cue phrases are considered as the units connectors. Each cue phrase has a specific action on determining the text units; we investigated these actions on the Arabic cue phrases and studied their effect on the discourse. The result is a set of actions that are associated with the cue phrases. Following are the actions that are used by the Arabic cue phrases to determine the text units of a discourse:

- *Normal*: add a unit boundary before the cue phrase.

- *Normal Then Comma*: add a unit boundary before the cue phrase and another unit boundary after the first occurrence of a *comma, semicolon or end of statement*. If a *comma* is met and followed by the cue phrases that correspond to *and "و" or or "أو"*, add unit boundary after the next occurrence of the three markers.

- *Dual*: same as the action *normal* if the cue phrase is not preceded by another cue phrase, action *normal then comma* is applied otherwise.
• *End:* add a unit boundary after the cue phrase.

• *Nothing:* bypass the cue phrase, this action is associated with a cue phrases that don't signal a rhetorical relations but help other cues in determining the text units.

In case of a cue phrase is followed by another cue phrase, we consider the first cue phrase as the unit boundary determiner if its action is not *nothing*, and the cue phrase that follows is by passed; the following example illustrates the case:

In this example, two cue phrases (underlined) exist in unit 2, the first one "أي أنه" signals the relation "تفسير (interpretation), and the second one "بسبب " تعليل (justification), the determiner will consider the first cue phrase as the unit boundary determiner, so it will apply its action, and this cue phrase will signal the rhetorical relation for unit 2.

In case of a cue phrase that has the action *nothing* was followed by another cue phrase that has different action, the second cue phrase will determine the text unit boundary based on its action; however, the unit determiner is put before the first cue phrase that has the action *nothing*. The following example highlights this context:

In this example, two cue phrase (underlined) exist in unit 2, the first one "هذا" has the action *nothing* and doesn't signal a discourse relation, and the second one "إذا" signals the relations *condition*, so the second cue phrase determines the rhetorical relation for unit 2, and it determines the text unit boundary according to its action, but the unit determiner will take care of the first cue phrase so that it will add the unit boundary before it instead of the second one. The same idea is applied in case if a cue phrase is followed by more than one cue phrase.
2.2 Relations Classification

The Arabic relations mentioned in (Tab 3.1) differ in term of the units they connect. Some relations connect two units, in this case they are called binary relations; in the other case, some relations connect one unit with one or more units, and are called bulky relations.

We noticed from the Arabic discourse that the relations "نتيجة" (result), "تمثيل" (example), and "قاعدة" (base) are bulky relations. The relation "نتيجة" (result) normally connects one unit as a result of one or more units. Consider the following example:

[ تعتبر هذه الأيام هي الأخيرة للفصل الدراسي الحالي.]1 [و من المتوقع حصول الكثير من الطلاب على نتائج جيدة هذا العام.]2 [نتيجة لذلك تستعد هيئة التدريس هذه الأيام لترتيب لحفل تكريم الطلاب المتفوقين.]2

We notice that unit1 and unit2 are parameters for the result mentioned in unit3.

The relation "تمثيل" (example) normally connects one unit with one or more units that are facts. Consider the following text fragment:

[من المعلومات أن الفواكه مصدر أساسي للفيتامينات الضرورية للجسم.]1 [و دائما ما ينصح الأطباء بتناول الفواكه بشكل مستمر.]2 [و من نعم الله على عبادة أن هذا المصدر متوفير طوال العام.]3 [فعلى سبيل المثال نجد البرتقال و النفاح متوفرين في الأسواق بشكل مستمر و هي من الفواكه الغنية بالفيتامينات الضرورية للجسم.]4

In this example, unit4 is related to all the units that came before; this is due to the fact that it is an example of the facts 1, 2, and 3.

The relation "قاعدة" (base) normally connects one unit as a base that one or more units are built upon. This example illustrates the relation:

[بناءً على ما رفعه مدير إدارة شئون الموظفين،]1 [فإن إدارة الشركة قررت ترقية جميع الموظفين الحاصلين على درجة التقييم 5 فما فوق.]2 [و منح هؤلاء الموظفين شهادات تقدير و تميز.]3

We notice that unit1 is related to both unit2 and unit3 as a base that these units are built upon.

The examples mentioned above showed that the three relations "نتيجة" (result), "تمثيل" (example) and "قاعدة" (base) might connect one unit with several units, and
because of this we will handle them depending on the position of the indicator in one of these two cases:

- If the cue phrase that signals the bulky relation comes in the first unit of the paragraph, this unit is related with all the subsequent units up to the end of the section.
- If the cue phrase that signals the bulky relation comes in a unit in the middle or the end of the paragraph, this unit is related to all the preceding units.

The other relations connect two units according to the position of the cue phrase.

2.3 Text Coherency

Text is principally coherent, that means a unit in the middle of the text might be related to one in the beginning. According to the building the rhetorical relations based on cue phrases that has been discussed in (section 2.1) the relation that will be built will relate adjacent units only, which will lead to a rhetorical representation that misses some important relations between units that are not adjacent. When dealing with building the rhetorical relations between the text units we recall what Marcu stated in (Marcu, 1997e) that “An accurate determination of elementary units of a text and of the relations that hold among them is beyond the current state of the art in natural language processing”.

To solve this problem, Marcu in (Marcu, 1997e, Marcu, 2000b) did a corpus analysis of the English cue phrases and assign a value called “Maximal distance” to each cue; this value holds the number of units found between the textual units that are involved in the rhetorical relation signaled by the designated cue phrase, and this number is manually assigned to each cue by studying the maximum number of units that come between the participant units of the rhetorical relation in the corpus analysis of the cue phrases. For example, the cue phrase “Although” has been assigned the value 5 for its participant units in the relation “elaboration”; this means that the relation “elaboration” is hypothesized between the unit that contains the cue phrase and all the four units that come before (Maximal distance-1 as specified by Marcu). This technique has some disadvantages which are:
• Complexity in determining the cue phrases: to determine the cue phrases that signal the rhetorical relations, we have to investigate the occurrence of each cue in several texts to determine the value of the maximal distance.

• Complexity adding more cue phrases: the relations and cue phrases are open lists; it means that they are subject for future expansion. Each time a relation is added, it is corresponding cue phrases need to be investigated in term of the maximal distance they have.

• The semantic of the cue phrases depend on the context: the semantic of a certain cue phrase is not fixed. A cue phrase in a certain context might connect certain number of units, whereas it connects different number of units in another context. Making each cue connect a fixed number of units for all the texts they appear in doesn’t reflect the fact that they depend on the context.

To avoid such disadvantages, we come up with a new approach. Our approach depends on the fact that the nuclearity relation among the text units is a transitive relation. This fact is based on (dilemma 3.1).

Dilemma 3.1:

\[
\text{If there are three units } u_1, u_2 \text{ and } u_3 \text{ such that:}
\]

\[
\text{rhet}_{-}\text{rel}_1 \text{ (name, } u_2, u_1) \text{ where } u_2 \text{ is satellite and } u_1 \text{ is nucleus, and}
\]

\[
\text{rhet}_{-}\text{rel}_2 \text{ (name, } u_3, u_2) \text{ where } u_3 \text{ is satellite and } u_2 \text{ is nucleus,}
\]

\[
\text{Then:}
\]

\[
u_1 \text{ is more salient than } u_3.
\]

We investigated the Arabic rhetorical relations, and we observed that in most cases, there is an implicit transitivity relation over the hypotactic Arabic rhetorical relations. Let’s have a look to the following example:

[لم يذهب خالد إلى السوق هذا اليوم;] [إنه لم يخرج من البيت] [سبب الأمطار الغزيرة.]
In the above example, unit2 has the relation “توكيد” (confirmation) with unit1, and unit3 has the relation “تتعلق” (justification) with unit2, and we can notice also that unit3 has the relation “تتعلق” (justification) of unit1 too. Finally, we hypothesize that that the hypotactic Arabic rhetorical relations are transitive; and we apply the following rules when we rhetorically parse the Arabic text:

- If \( \text{rhet}_1 (\text{rel}_1, s_1, n_1) \), and \( \text{rhet}_2 (\text{rel}_2, s_2, n_2) \) such that \( n_2 = s_1 \) → new relations \( \text{rhet}_3 (\text{rel}_2, s_2, n_1) \)
- If \( \text{rhet}_1 (\text{rel}_1, s_1, n_1) \), and \( \text{rhet}_2 (\text{rel}_2, s_2, n_2) \) such that \( n_1 = s_2 \) → new relations \( \text{rhet}_3 (\text{rel}_1, s_1, n_2) \)

Since paratactic relations have both of their spans as nucleus, transitivity is not applied on them.

In some situations, there is no cue phrase connector between two units; in this case the relation is hypothesized between these units using a technique called word co-occurrences. In this technique the parser parses the units that don't have a cue phrase connector and count the number of the words that exist in the participant units; if the number of a certain word existence exceeds a certain threshold, the second unit is consider to have the relation "تفسير" (explanation) with the first one. If the number of the co-occurred words doesn't exceed the threshold, the relation "عطف" (joint) is hypothesized between the units. Notice that when counting the co-occurred words, the words are not taken as they are since two similar words might have different morphemes; the words are stemmed and then compared instead. The problem is that in Arabic language we lack an accurate stemming algorithm, but we developed a simple stemming algorithm that is suitable for word comparison but it is not 100% accurate. The algorithm is explained in (Fig 3.2). When the stemmed words are compared, the comparison algorithm checks the two words; if they have similar three or more letters it then considers them as similar words, or it will consider them as not similar otherwise.
Input: A word.
Output: A stemmed word.

BEGIN
if word.length < 3
  return word;  // don't stem, consider it as Arabic letter
if first two letters are "لا"
  remove them;
if word length < 2
  return word; // don't stem, consider it as Arabic letter
if last letter(s) are in the set {"ه", "ه", "ه", "ت", "ون", "ن", "ان", "ات", "ون", "ن"}
  remove it/them and then return the new word;
END.

Fig 3. 2. Arabic word stemming algorithm.

2.4 The Most Suitable RS-tree for Arabic Text Summarization

The generated summary based on the RS-tree depends mainly on the tree that was selected since there may be several trees generated for a rhetorically parsed text. The variety of the trees results from the relation hypothesis among the text spans. Since there isn’t a unique technique to extract a relation between two spans, the algorithm hypothesizes a set of relations which leads to various RS-trees. Let’s consider the following text:

[بعض الناس عندهم القابلية على (الفلق)2[أي الخوف المفرط]2 إذا تعرضوا للضغوط.]

In this text, unit two has the relation "تفسير" (interpretation) with unit one, this relation was extracted from the cue phrase (أي); whereas, unit three has the relation "شرط" (condition) since it has the cue (إذا); unfortunately, this technique can't decide in a unique way, to which unit a relation is related to! The technique always assumes that each two adjacent unit are related to each other, so it will hypothesize that unit three has the relation "شرط" (condition) with unit two, and since unit two is related to unit one and it is the satellite part, unit three is hypothesized to have the relation "شرط" (condition) with unit one too. The final extracted relations will be:

- rhet_rel (تفسير, 2, 1)  // interpretation
- rhet_rel (شرط, 3, 2)  // condition

42
If we want to build the RS-tree for this text based on the above relations, we will end-up with two trees shown in (Fig 3.2).

![Two RS-trees generated for one text.](image)

These two trees will give different results in text summarization since level one is different. We have to decide which tree is the most suitable tree for text summarization. Suppose we wants to summarize this text and we decided to go to level one; if we follow the extracted relations logically, we will select tree(b) since unit three is satellite for unit two and unit one. But if we look at the two generated summaries (Fig 3.3) we will notice that tree (a) gives a better result!

**Summary generated from tree(a)**

بعض الناس عندهم القابلية على (القلق) إذا تعرضوا للضغط.

**Summary generated from tree(b)**

بعض الناس عندهم القابلية على (القلق) أي الخوف المفرط.

![Two summarized texts generated from two RS-trees.](image)

So following the logic of the extracted relations to select the RS-tree doesn't guarantee that the select tree is the most suitable one for text summarization, since the
algorithm doesn't extract the relations precisely, it instead hypothesize a set of relations that one or more is applied. So we need a technique to select the most suitable tree for text summarization.

In (Marcu, 1997e), Marcu stated that the best discourse trees are usually those that are skewed to right. He justified his observation by the fact that text processing is a left to right process, and he mentioned another justification based on his experiment that people write text so that the most important units go first, and the more text writers add, the more elaborate on the text that went before.

In order for us to determine the most suitable RS-tree for Arabic text summarization, we studied the Arabic corpus and the way that Arabic writer write. Writers mainly want to convey a message to readers. This message is mentioned as several facts; however, the study we did on the Arabic corpus shows that writers tend to mention those facts in sequence, and each fact is followed by statements that support it. The style of the Arabic corpus could be represented as in (Fig 3.4).

![Fig 3.5. Arabic writing style observed from the Arabic corpus study.](image)

The RS-trees that are generated to a given text could be classified into three categories (Marcu, 2000b):

- Skewed to right.
- Balanced.
- Skewed to left.

In the skewed to right trees the units that come at the beginning of the text will have the highest possibility to be escalated up to the high levels in the tree –tree (b) in (Fig 3.2) is an example, so the supportive statements for the facts that come at the beginning of the text might be escalated over the facts that follow. In the skewed to left trees the units that come at the end of the text have the highest possibility to be escalated
up to the high level, so the supportive statements for the facts that come at the end of the
text might be escalated over the facts that come at the beginning of the text. When the
tree tends to be balanced, each fact is related with its supportive statements leading to the
escalation of the facts over the supportive statements. Consequently, the more the tree is
balanced, the more suitable it is for text summarization.

One more reason for considering the more balanced trees as the most suitable for
text summarization is that, in the most balanced trees the competition between the
sentences to be escalated is higher leading to have the most important units at the top
levels of the tree. In (Fig 3.5), we observe that in tree (c), unit one was escalated directly,
the same case is noticed concerning unit four in tree (b), whereas in tree (a) unit one
competed with unit two and unit three competed with unit four.

Based on the style of the Arabic writing, the tree that is most skewed to right
among the most balanced trees is the most suitable. This is due to the style of the Arabic
writing explained in (Fig 3.4); we observe that in most cases the fact comes before the
supportive statements; from this observation and from the experiment shown in the next
section, we consider the trees that are skewed to right are better than the ones that are
skewed to left. Consequently, if we have more than one tree which are most balanced, we
select the one that is skewed to right among them.

Fig 3. 6. The three categories of the RS-tree.
We have design an algorithm for the rhetorical parsing of the Arabic text which is shown in (Fig 3.6), and then implemented it using java programming language. The program rhetorically analyzes the Arabic texts and generates all the possible RS-trees for that text, and then we extract the summary for the generated texts by going to level one in the generated trees (we consider root as level zero); finally, we evaluated the generated summaries and got the following result.

− 10% of the trees that are skewed to left were the most suitable.
− 20% of the trees that are skewed to right were the most suitable.
− 70% of the trees that are most balanced were the most suitable.

As the result shows we noticed that in most cases the most balanced trees gave the most convenient result.

\begin{verbatim}
Input  A text
Output RS-tree for the input text

BEGIN
  I Determine the set of all cue phrases in the input text
  II Determine the elementary units in the input text based on the cue phrase
determined in step I and text markers
  III Hypothesize the rhetorical relations between the elementary units based
on the cue phrases determined in step I
  IV Build all the RS-trees for the given text based on the hypothesized relations
in step II
  V Select the most suitable RS-tree that best represent the input text and suits
your application

END.
\end{verbatim}

\textbf{Fig 3.7.} Rhetorical parsing algorithm.

Following are some examples on the technique.

\textbf{Example 3.5:}

This example shows the result of the program we implemented when it is given the following text:
The program hypothesized the following rhetorical relations:

- `rhet_rel` (تفصيل, 2, 1)  // explanation
- `rhet_rel` (تفصيل, 3, 1)  // explanation
- `rhet_rel` (تفصيل, 3, 2)  // explanation
- `rhet_rel` (توثيد, 4, 1)  // confirmation
- `rhet_rel` (توثيد, 4, 2)  // confirmation
- `rhet_rel` (توثيد, 4, 3)  // confirmation

The RS-trees in (Fig 3.7) were generated.

![Fig 3. 8. Three RS-trees generated by the program for the text in example 3.5.](image)

Summary Extracted from tree (a):

The advice here is that investors should consider different types of stocks and bonds, and the program hypothesizes the following relations:

- `rhet_rel` (تفصيل, 2, 1)  // explanation
- `rhet_rel` (تفصيل, 3, 1)  // explanation
- `rhet_rel` (تفصيل, 3, 2)  // explanation
- `rhet_rel` (توثيد, 4, 1)  // confirmation
- `rhet_rel` (توثيد, 4, 2)  // confirmation
- `rhet_rel` (توثيد, 4, 3)  // confirmation

The RS-trees in (Fig 3.7) were generated.

![Fig 3. 8. Three RS-trees generated by the program for the text in example 3.5.](image)

Summary Extracted from tree (a):

The advice here is that investors should consider different types of stocks and bonds, and the program hypothesizes the following relations:

- `rhet_rel` (تفصيل, 2, 1)  // explanation
- `rhet_rel` (تفصيل, 3, 1)  // explanation
- `rhet_rel` (تفصيل, 3, 2)  // explanation
- `rhet_rel` (توثيد, 4, 1)  // confirmation
- `rhet_rel` (توثيد, 4, 2)  // confirmation
- `rhet_rel` (توثيد, 4, 3)  // confirmation

The RS-trees in (Fig 3.7) were generated.
Summary Extracted from tree (b):

We notice from (example 3.5) that the balanced tree –tree (b), produced the best result among the other two trees.

Summary Extracted from tree (c):

We notice from (example 3.5) that the balanced tree –tree (b), produced the best result among the other two trees.
Chapter 4: A Pattern-Oriented System Model

In this chapter we propose a pattern-oriented system model for the Arabic text summarization using rhetorical structure theory. The aim of using the pattern-oriented approach is to make the system flexible and maintainable since it is an infrastructure and subject for future improvement and expansion.

The pattern was stated in (Gamma et al., 1995) as following:

"Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice".

Another definition was stated in (Buschmann et al., 1996) as following:

"Each pattern is a three-part rule, which expresses a relationship between a certain context, a problem, and a solution".

It is also stated in (Gamma et al., 1995) that a pattern has four essential elements:

1. The pattern name is a handle we can use to describe a design problem, its solutions, and consequences in a word or two.

2. The problem describes when to apply the pattern. It explains the problem and its context. It might describe specific design problems such as how to represent algorithms as object. It might describe class or object structures that are symptomatic of inflexible design.

3. The solution describes the elements that make up the design, their relationships, responsibilities, and collaboration.

4. The consequences are the results and trade-offs of applying the pattern.

Pattern-oriented approach is an efficient way to design software infrastructures since it provides efficient design templates that could be built upon.
In this chapter we first state the Arabic text summarization using rhetorical structure theory system requirements and aspects. Then we propose a model for the system architecture. Finally we propose some pattern-oriented design ideas for the system and those ideas have been implemented.

1 System Aspects and Requirements

The system has to follow the rhetorical parsing process which consists of:

- Determine the elementary units.
- Hypothesize the rhetorical relations between the elementary units determined in the previous step.
- Build all possible RS-trees.
- Select the most suitable RS-tree.

There are some constraints on designing the system. Some processes are implemented in different algorithms, the text might be input in different formats (word doc, html, plain text … etc), the system should take care of the different Arabic text encodings (Unicode, Cp1256 … etc), and the complete summarization process should be designed in a way that makes it flexible.

The list of rhetorical relations and the list of cue phrases are opened lists. Any change or future extension to these list should not affect the neither the design nor the code. Therefore, the system should be design in away that allows updating the two lists in a simple and flexible way.

The design of the data model (for the relations, and cues) should take into account the data consistency and should not employee any constraints on the size of the tables since the tables are subject of expansion.

Following sections presents the design of the system taking into account the above constraints and aspect.
The system is divided into three main units as shown in (Fig 4.1), the Arabic text summarizer, the RST processor, and the shared data.

*Fig 4.1. System architecture of the Arabic text summarizer based on RST.*

The process goes into several steps:

- *Arabic Summarizer* parses the text and then notifies the *Elementary Units Determiner* that the text is ready for processing.
• **Elementary Units Determiner** gets the text, divides it into elementary units, and then notifies the **Rhetorical Relations Hypothesizer** that the elementary units are ready for processing.

• **Rhetorical Relations Hypothesizer** gets the elementary units, hypothesizes the rhetorical relations among them, and then notifies the **RS-tree Builder** that the relations are ready.

• **RS-trees Builder** gets the relations, builds all the valid RS-trees, and then notifies the **RS-tree Selector** that the RS-trees are ready.

• **RS-trees Selector** selects the most suitable RS-tree for Arabic text summarization.

3 System Packages

The modules are distributed over packages according to their functional behaviors. The system packages can be inferred from the system architecture shown in (Fig 4.1). The system contains three packages; the first package "Arabic text summarizer" contains the Arabic summarizer class; this class is the main Arabic text summarizer that uses the rhetorical structure theory technique. This package is supposed to hold any other Arabic text summarizer classes that might use other techniques. The second package "RST processor" contains the classes that are related to the RST. Finally, the package "Shared data" contains the classes for the data that is needed by the complete processing.

Currently, the packages hold the classes shown in (Fig 4.1), but they are subject of future expansion. The package "RST processor" might be extended to hold additional classes needed for further processing. The growth of the package "Shared data" depends on the growth of the other two packages. (Fig 4.2) shows the distribution of the classes of the three packages.
As shown in (Fig 4.2) the package "Arabic text summarizer" depends on the package "RST processor" since it the summarizer is based on the RST; it also depends on the package "Shared data" to feed data on it, and then get the final result. Package "RST processor" depends on the package "Shared data" where each process gets the data it needs, and then deposits the result for the following process.

The classes involved in the diagram shown in (Fig 4.2) interact to form the complete system. The following sections demonstrate some software patterns that could be used when developing such systems. Each pattern is aimed to solve a certain considerable design issue.
4 System Patterns

4.1 Chain of RST Processes Pattern

RST processes shown in (Fig 4.1) are executed consecutively. Each process is executed and then triggers the following one. (Fig 4.3) shows a diagram of the complete RST process behavior.

![Diagram of RST processes chain](image)

Fig 4.3. RST processes chain.

The problem is that when this chain is modified for any reason (adding more process, removing any process … etc), some processes that are exposed to this change might be changed to.

This pattern limits the changes of the processes in case of any change is made to the chain.

Solution

Decouple the processes from each others, and let the client decide the sequence and the classes that participate in the chain.

The structure shown in (Fig. 4.4) explains the solution. The ArabicTextSummarizer is a client of RSTprocess. The client determines the processes it needs and their sequence (as it will be demonstrated in the code example). Each process executes its tasks, and then it lets the abstract class triggers the next process according to the configuration of the client. New process could be added, or existing process could be deleted without affecting the other processes. Any change is made to any process, other processes are not aware of.
Example Code

```java
public abstract class RSTprocess {
    private RSTprocess nextProcess;
    public abstract void run();
    public void runNextProcess () {
        nextProcess.run();
    }
    public void setNextProcess (RSTprocess process) {
        nextProcess = process;
    }
}

public class ElemUnitsDeterminer extends RSTprocess{
    public void run () {
        // execute the process of determining
        // the elementary units
        super.runNextProcess();
    }
}
```

Fig 4.4. Chain of RST processes pattern class structure.
public class RhetRelHypothesizer extends RSTprocess{
    public void run () {
        // execute the process of determining
        // the elementary units
        super.runNextProcess();
    }
}

public class RStreesBuilder extends RSTprocess{
    public void run () {
        // execute the process of determining
        // the elementary units
        super.runNextProcess();
    }
}

public class RStreeSelector extends RSTprocess{
    public void run () {
        // execute the process of determining
        // the elementary units
        super.runNextProcess();
    }
}

public class ArabicTextSummarizer {

    private RSTprocess rstProc;

    public ArabicTextSummarizer() {
        RSTprocess process1 = new RStreeSelector ();
        RSTprocess process2 = new RStreesBuilder ();
        process2.setNextProcess(process1);

        process1 = process2;
        process2 = new RhetRelHypothesizer ();
    }
}
process2.setNextProcess(process1);

rstProc = new ElemUnitsDeterminer();
rstProc.setNextProcess(process2);
}

public void execute () {
    Text text = new Text();
    SharedObject sharedData = new SharedObject();
    // parse the text
    // then deposit the parsed text into the shared
    // object so that ElemUnitsDeterminer gets it from there
    sharedData.setText (text);
    rstProc.run();
}

The above java code shows how this pattern could be implemented; it shows all the participant classes as well as how the client interacts with the RST process chain.

Consequences

The consequences of applying this pattern are the following:

- RST processes are totally decoupled and are not aware about each others.
- Client is responsible for instantiating the RST processes it needs.
- Flexibility in adding more processes when they are need, deleting any existing process, and modifying the behavior of any process.
- All the RST processes have common shared interface.
- Client is not aware about the internal process of the RST.
4.2 Multi-Relations Hypothesizers Pattern

As described in (chapter 2, and chapter 3), the rhetorical relations are hypothesized using several techniques. In our case, we use two techniques (cue phrases, and word co-occurrences). In the summarization process, we need to move between the two techniques dynamically. Moreover, new techniques should be added in a flexible way.

This pattern is the answer of the above problem. It makes a common interface for all the relations hypothesis techniques, and let the client decide which technique it wants to use. The class diagram shown in (Fig 4.5) describes the structure of this pattern.

![Class Diagram](image_url)

**Fig 4.5.** Multi-Relations hypothesizers pattern class structure.

As shown in (Fig 4.5), the client (ArabicTextSummarizer in our case) has the responsibility of deciding which technique it wants to use. It can change the technique dynamically at any time in the program lifetime. When new techniques are added, they can be used by the client easily. This pattern is very helpful since there are many techniques in hypothesizing the rhetorical relations. It is the responsibility of the system developers to decide which techniques they want to use, and the pattern provides them a maintainable way of employing any technique and then update it at any time in the system life cycle.
Example Code

We did some changes to the classes ArabicTextSummarizer, and RhetRelHypothesizer to suite the new pattern.

public abstract class RhetRelHypothesizer extends RSTprocess{
    public abstract void run ();
}

public class BasedOnCuePhrases extends RhetRelHypothesizer{
    private void hypothesize () {
        // hypothesize relations based on cue phrases
    }

    public void run () {
        hypothesize ();
    }
}

public class BasedOnWordCooccurrence extends RhetRelHypothesizer{
    private void hypothesize () {
        // hypothesize relations based on cue phrases
    }

    public void run () {
        hypothesize ();
    }
}

public class ArabicTextSummarizer {

    private RSTprocess rstProc;

    public ArabicTextSummarizer(RhetRelHypothesizer tech) {
        RSTprocess process1 = new RStreeSelector ();
        RSTprocess process2 = new RStreesBuilder ();
    }
process2.setNextProcess(process1);

techn.setNextProcess(process2);

rstProc = new ElemUnitsDeterminer();
rstProc.setNextProcess(tech);
}

public void execute () {

    Text text = new Text();
    SharedObject sharedData = new SharedObject();
    // parse the text
    // then deposit the parsed text into the shared
    // object so that ElemUnitsDeterminer gets it from there
    sharedData.setText(text);
    rstProc.run();
}

public static void main (String args[]) {

    for (int i = 0; i < numberOfTexts; i++) {
        RhetRelHypothesizer hypo = null;
        if ((i%2) == 0)
            hypo = new BasedOnCuePhrases();
        else
            hypo = new BasedOnWordCooccurrence();
        ArabicTextSummarizer ats =
            new ArabicTextSummarizer(hypo);
        ats.execute();
    }
}
}
As shown in the example, the technique is selected dynamically. For odd numbers it uses the technique *BasedOnWordCooccurrence*, and for even numbers it uses the technique *BasedOnCuePhrases*. It also allow adding a new technique when needed and then instantiate it by passing it as a parameter to the *ArabicTextSummarizer* constructor.

**Consequences**

The consequences of applying this pattern are the following:

- It allows selecting the hypothesis technique dynamically.
- It allows adding more technique without touching the other techniques, and without major changes to the client.
- It provides a common interface for all techniques.

### 4.3 Multi-RS-tree Selectors Pattern

As described in (chapter 2 and chapter 3) there are many ways of selecting the most suitable RS-tree for text summarization. As it has been stated that Marcu suggests selecting the RS-tree that is most skewed to right for English corpus. As our study on Arabic corpus, we observed that the most suitable RS-tree for text summarization those that are most balanced. These two observations shows that there are several ways for selecting the RS-tree depending on the language, application (text summarization in our case), and the observation of the developer.

The above description shows the need of making a design in such a way that allows the dynamic change of the technique of selecting the RS-tree, and allows adding more technique, or changing or deleting an existing technique. The class diagram shown in (Fig 4. 6) gives the structure of the pattern that provides the solution.
Fig 4. Multi-RS-tree selectors pattern class structure.

Example Code

```java
public abstract class RStreeSelector extends RSTprocess{
    public abstract void run();
}

public class MostSkewedToRight extends RStreeSelector {
    private void select () {
        // select the tree that is most skewed to right
    }

    public void run () {
        select ();
    }
}

public class MostBalanced extends RStreeSelector {
    private void select () {
        // select the tree that is most balanced
    }
}
```
public void run () {
    select ();
}
}

public class ArabicTextSummarizer {

    private RSTprocess rstProc;

    public ArabicTextSummarizer(RhetRelHypothesizer hypothesizer,
                                RStreeSelector selector) {
        RSTprocess process = new RStreesBuilder ();
        process.setNextProcess(selector);

        hypothesizer.setNextProcess(process);

        rstProc = new ElemUnitsDeterminer ();
        rstProc.setNextProcess(hypothesizer);
    }

    public void execute () {

        Text text = new Text ();
        SharedObject sharedData = new SharedObject ();
        // parse the text
        // then deposit the parsed text into the shared
        // object so that ElemUnitsDeterminer gets it from there
        sharedData.setText (text);
        rstProc.run();
    }

    public static void main (String args[]) {

        for (int i= 0; i < numberOfTexts; i++) {
            RhetRelHypothesizer hypo = null;

RStreeSelector select = null;
if ((i%2) == 0) {
    hypo = new BasedOnCuePhrases ();
    select = new MostSkewedToRight ();
}
else {
    hypo = new BasedOnWordCooccurrence ();
    select = new MostBalanced ();
}
ArabicTextSummarizer ats =
    new ArabicTextSummarizer (hypo, select);
ats.execute();
}
}

As shown in the example, the technique is selected dynamically. For odd numbers it uses the technique MostBalanced, and for even numbers it uses the technique MostSkewedToRight. It also allow adding a new technique when needed and then instantiate it by passing it as a parameter to the ArabicTextSummarizer constructor.

Consequences

The consequences of applying this pattern are the following:

• It allows selecting the RS-tress selector technique dynamically.

• It allows adding more technique without touching the other techniques, and without major changes to the client.

• It provides a common interface for all techniques.
5 System Scenario

The behavior of the system that has been described in (section 2), and the objects lifetime are described by the scenario expressed in the sequence diagram shown in (Fig 4.7).

Fig 4.7. Arabic text summarization system scenario.
6 System Data Model

The system uses some data to perform its process. The data is related mainly to the cue phrases that are being employed in the process. As described in (chapter 3), the cue phrases have some features that are used to signal the relations between the text units. Those features are stored in a relational database. (Fig 4.8) shows the cue phrase entity, and (Fig 4.9) shows the table as a result of mapping the entity in (Fig 4.8) into a relational model.

Fig 4.8. Arabic text summarizer data model.

Two entities are shown in (Fig 4.8), the first one is Elementary_units_determiner which represents the cue phrases that determine the elementary units of a text. the second entity is the Relations_signaler which represents the cue phrases that signals the rhetorical relations among the text units. The relations WorksAs relates the cue phrases that determine the elementary units with others that signal the rhetorical relations in such a way that the cue phrases that determine the elementary units signal rhetorical relations and vise versa. One cue phrase which determines elementary units might signal more than one relation depending on its position on the text; therefore, the relation between the two entities is 1:N.
Fig 4.9. Arabic text summarization data tables.
Chapter 5: Evaluation

To evaluate the approach this thesis introduces, we have implemented software based on this approach. The implementation was done using JAVA programming language, and Access database. We collected different texts from articles taking about different subjects (politics, society, sports, education and computer). We distribute those articles to judges from different disciplines; we asked them select the most important units in the articles, and then we compare them with the result of the auto-summarization software. The collected articles were classified into two groups: small size articles (10-15 lines), and medium size articles (30-35 lines). The reason of not including large texts in the evaluation is that the complexity of the proof-theoretic account based algorithm that builds all the valid RS-trees is high as stated in (Marcu, 1997e); for large texts the result might take hours to be generated. We will attempt to predict the behavior of the software for large text from the result of the sample we collected. (Tab 5.1) shows the result of the evaluation of the small size articles; (Tab 5.2) is the result of the evaluation for the medium size articles.

<table>
<thead>
<tr>
<th>Summarizer</th>
<th>Text#</th>
<th>Manually Selected</th>
<th>Automatically Selected</th>
<th>Matched</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Judge1</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>100%</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>100%</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>50%</td>
</tr>
<tr>
<td>Judge2</td>
<td>11</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>50%</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>100%</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>100%</td>
</tr>
<tr>
<td></td>
<td>17</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>50%</td>
</tr>
<tr>
<td></td>
<td>21</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>Judge3</td>
<td>18</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>50%</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>60%</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>100%</td>
</tr>
<tr>
<td></td>
<td>27</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>100%</td>
</tr>
<tr>
<td></td>
<td>29</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>60%</td>
</tr>
<tr>
<td>Judge4</td>
<td>7</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>60%</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>50%</td>
</tr>
</tbody>
</table>
Tab 5.1. Result of auto-summarization software for small size articles.

<table>
<thead>
<tr>
<th>Summarizer</th>
<th>Text#</th>
<th>Manually Selected</th>
<th>Automatically Selected</th>
<th>Match</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Judge1</td>
<td>4</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>100%</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>50%</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>50%</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>100%</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>100%</td>
</tr>
<tr>
<td>Judge2</td>
<td>1</td>
<td>8</td>
<td>2</td>
<td>2</td>
<td>100%</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>100%</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>100%</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>50%</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>50%</td>
</tr>
<tr>
<td>Judge3</td>
<td>12</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>30%</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>50%</td>
</tr>
<tr>
<td>Overall Precision</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>62%</td>
</tr>
</tbody>
</table>

Tab 5.2. Result of auto-summarization software for medium size articles.

The accuracy curve based on the result of the two experiments is shown as in (Fig 5.1). Y axis represents the precision, and X axis represents the text size. The reason of the small gap between the precisions of the two groups (small size articles, medium size articles) is that when the size of the text is increased, more important information are mentioned, and then the size of the summary is increased accordingly. This results the match between the auto-summarization software and the manual summarization.
Fig 5.1. The behavior of the auto-summarization software.
Appendix I: Cue Phrases

(Tab 6.1) lists the cue phrases that we used in the experiment.

<table>
<thead>
<tr>
<th>Cue Phrase</th>
<th>Relation</th>
</tr>
</thead>
<tbody>
<tr>
<td>إذن</td>
<td>توكيد</td>
</tr>
<tr>
<td>إلا أن</td>
<td>تفصيل</td>
</tr>
<tr>
<td>أو</td>
<td>Nothing</td>
</tr>
<tr>
<td>بل</td>
<td>توكيد</td>
</tr>
<tr>
<td>بمعنى أن</td>
<td>تفسير</td>
</tr>
<tr>
<td>ثم</td>
<td>ترتيب</td>
</tr>
<tr>
<td>غير أن</td>
<td>تفصيل</td>
</tr>
<tr>
<td>لذا</td>
<td>نتيجة</td>
</tr>
<tr>
<td>لقد</td>
<td>Nothing</td>
</tr>
<tr>
<td>لكن</td>
<td>ابتدأ</td>
</tr>
<tr>
<td>لو</td>
<td>شرط</td>
</tr>
<tr>
<td>من ثم</td>
<td>ترتيب</td>
</tr>
<tr>
<td>نتيجة لذلك</td>
<td>نتيجة</td>
</tr>
<tr>
<td>نتيجة لهذا</td>
<td>نتيجة</td>
</tr>
<tr>
<td>نستنتج</td>
<td>نتيجة</td>
</tr>
<tr>
<td>و</td>
<td>Nothing</td>
</tr>
<tr>
<td>إذا</td>
<td>شرط</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td>إن</td>
<td>شرط</td>
</tr>
<tr>
<td>بسبب</td>
<td>تعليل</td>
</tr>
<tr>
<td>بما أن</td>
<td>قاعدة</td>
</tr>
<tr>
<td>بناءاً على</td>
<td>قاعدة</td>
</tr>
<tr>
<td>حيث</td>
<td>تفسير</td>
</tr>
<tr>
<td>على سبيل المثال</td>
<td>تمثيل</td>
</tr>
<tr>
<td>لأن</td>
<td>تعليل</td>
</tr>
<tr>
<td>لاسيما</td>
<td>تفصيل</td>
</tr>
<tr>
<td>مثلا</td>
<td>تمثيل</td>
</tr>
<tr>
<td>خاصة</td>
<td>Nothing</td>
</tr>
<tr>
<td>أي</td>
<td>Nothing</td>
</tr>
</tbody>
</table>

**Tab 6.1.** Cue phrases used in the experiment.
Appendix II: Example of the Auto-Summarization Software

Aقبل الصيف علينا للكشف أننا لم نف بوعودنا التي قطعناها بعد معاونانا من ترتيبات السفر التي قمنا بها في الوقت
الضائع من صيف العام الفاضل ... تلك الترتيبات (أو عدمها) التي ترفع الضغط والسّكر وغيرهما من مؤشرات
أبدناها، وتخليق المشاكل المالية والاجتماعية وغيرها. عندما لا نخطط لإجازتنا نخسر كثيراً وعلى كل المستويات، ثم
نطلق الوعود للعام التالي لنفاجأ بوصوله بدون تنفيذها.

منا من يفاجأ بأنه لم يخصص مساحة ميزانية للسفر فيتدين من بطاقات البنوك أو غيرها، ثم يقضي أغلب أشهر
العام التالية للإجازة يسّدد ذلك الدين. ومنا من تجبره أوضاع حجوزات الطيران والتأشيرات، على السفر إلى مدن لم
تكيئن فجأة رغبتنا الأولي ولا الثانية ولا الثالثة.

كما أننا ندفع أعباء أعلى لتذكر السفر والفنادق، ونضطر للشتات في مقاعد الطائرات والفنادق، لأننا نجمع
المقاعد والغرف المُلغي حجزها هنا وهناك في آخر لحظة. ومنا من يفاجأ بأن موعد رحلته سابق صدر تأشيرته,
وربما أضع عليه ذلك قيمة التذكرة أو جزء منها أو اضطره لدفع غرامة.

السفر يحتاج إلى تخطيط لبحق أهدافه المرجوة بأسعار معقولة، ودون أن يكون هو مصدر مشاكل لطالبيه. حين
تخطيط تختار المدن التي تزورها، وتتأكد من صلاحية جوازك في وقت مبكر، وتحصل على التأشيرات في الوقت
 المناسب، وتم حجوزات الطيران والسكن بشكل سهل وبأسعار أفضل، وتحت أيام المغادرة والعودة. كما أنك
تستطيع تغيير مواقع الإجازات بين الدول وبين الداخل والخارج. حين تخطيط تكون مصاريف إجازاتك في حدود
إمكانتك، ويكون ذلك من أسباب المتعة الحقيقية في الإجازة.

وcuda ما لا تخطط تكون إجازاتنا على كيف عرضت، تزعجنا في ترتيبها، وتأثنا إلى حيث نرغب ولا نرغب وبأسعار
مرتفعة. كما أنها تكون مليئة بالمخاطر غيرة السارة. ومنا، مع الأسف، من يدرك تلك المشكلة ويتعامل معها بشكل
غير مقبول، فيرحب مقاعد في أكثر من رحلة ولاكثر من وجهة، على أمل أن يوفق ذلك أو يلغي في وقت لاحق ثم
يتجاهلها أو ينساها، فلا يستفيد منها هو ولا غيره ويضع فرصة تسويقها في الوقت المناسب على شركات
الطيران.
على آية حلال ليست كل السفرات قابلة للتخطيط والترتيب المبكر، فهناك سفرات مفاجئة لا يمكن التخطيط لها كسفرات العلاج المفاجئة. كفانا الله وإياكم شرها، وتلك لا ينطبق عليها شيء مما ورد. جعل الله كل سفراتها قابلة للتخطيط ووقفنا لفيام به.

Auto-generated Summary:

أقبل الصيف علينا لكشف آتنا لم نف بوجودنا التي قطعناها بعد معاناتنا من ترتيبات السفر التي قمنا بها في الوقت الضائع من صيف العام الفات. عندما لا نخطط لياجتنا نخسر كثيرا وعلى كل المستويات. على آية حلال ليست كل السفرات قابلة للتخطيط والترتيب المبكر، فهناك سفرات مفاجئة لا يمكن التخطيط لها كسفرات العلاج المفاجئة. كفانا الله وإياكم شرها، وتلك لا ينطبق عليها شيء مما ورد.
References


