CHAPTER 1

1.1 BACKGROUND

Sentence compression is the task of compressing a long sentence into a short one. It retains the important contents and in the meantime it generates grammatical short sentences. There are several applications in which the generation of compressed sentences is useful, such as television closed captions, visually disabled & portable devices. However, small screen size and storage capacity act as hurdles to access such huge amount of information, People with language disabilities like aphasia have difficulty reading long and complex sentences. By using sentence compression, we can display small meaningful sentences which enable them to read without effort.

Sentence compression is also useful as a pre-processing tool (Chandrasekar, R. and Doran,C. and Srinivas.B) as for the following Natural language processing (NLP) applications:

Parsing: Syntactically complex sentences are likely to generate a large number of parses, but simpler sentences lead to faster parsing and less parse ambiguity.

Machine Translation (MT): since MT algorithms are suffer of producing low quality outputs (translated sentences) from long sentences, sentence compression techniques aim to improve their performance qualities.

Information Retrieval: sentence compression algorithms are used to improve retrieval process (example search engines).

1.2 PROBLEM STATEMENT

Proposing a sentence compression algorithm that conserves sentence cohesiveness and coherence is considered a difficult task in natural language processing. This research aims to investigate the performance of a new algorithm for compressing the long sentences based on semantic role labeling (SRL).

1.3 RESEARCH OBJECTIVES

The main goal of this study is to investigate the integration of semantic information for enhancing a problem of sentence compression. The following are the sub-goals:

1. To investigate whether the integration of semantic role labeling (SRL) may produce cohere and grammatical sentences.
2. To investigate whether the integration of the SRL technique captures the most salient pieces of information in original sentence.

1.4 RESEARCH QUESTION

This research comes to answer the following questions:

1. Is our improved method efficient to generate cohere and grammatical sentence?

2. Is our improved method compresses long sentence and retains the most of salient pieces of information found in the original sentences?

1.5 RESEARCH HIPOTHISIS

This research found to test the following hypothesis:

"Compressing long sentence based on semantic information is more advantageous & useful than using” lexical information”.

1.6 RESEARCH SCOPE

The scope of this research to propose an improved method based on semantic role labeling. Ziff Davis dataset is used to train and test our proposed method. In addition SENNA SRL extractor is used to extract semantic roles for each input sentence.

1.7 RESEARCH SIGNIFICANCE

Sentence compression concept is considered important and useful for several applications. It is also:

1. Keeps space with the program in television closed caption.
2. Improve the efficiency of screen readers for the visually disabled.
3. Delivers compressed content to portable devices.
4. Helps people with language disabilities like aphasia.

Sentence compression is also useful for many natural languages processing application such as: texts parsing, machine Translation, information retrieval, and Text summarization.

1.8 RESEARCH STRUCURE

This thesis composed of five chapters. Chapter one presents background about the problem, objectives of the research, Scope, research significance and hypothesis. Then Chapter Two introduces Literature Review and related works, it presents the definition of sentence compression, semantic role labeling, significance of semantic role labeling. Chapter Three presents the methodology of **the new algorithm**. Chapter Four discusses the implementation results. Chapter five concludes the thesis findings.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter presents a brief introduction to natural language processing (NLP) and their applications such as (information retrieval, text summarization, etc.). NLP is an important and essential topic related to Human Computer Interaction (HCI). This chapter provides an overview about the existing methods for NLP in sentence compression and review a large number of related work using these methods in order to find a new method to investigate sentence compression. However, the main aim of this chapter is to present a new method to investigate the sentence compression using Semantic Role Labeling (SRL). This chapter found two studies which are most related to our study and critically were analyzed. The rest of this chapter contains 10 Sections organized as follows. Section 2.2 introduces NLP. An overview about information Retrieval is presented in Section 2.3. Text Summarization is reviewed in section 2.4. Section 2.5 presents a brief introduction to sentence compression. Section 2.6 reviews some works used noisy-channel for sentence compression. Then, Section 2.7 gives an introduction to semantic role labeling with some related works. Whereas, Section 2.8 explains the concept of the datasets and selected one to be used in this research. Sentence similarity measures are discussed in Section 2.9. And lastly, the chapter is summarized in Section 2.10.

2.2 Introduction to Natural Language Processing

The communicating between human and non-human devices is very complex task. To reduce the complexity associated with it, the new ways is needed to investigate this problems and to develop new method such as NLP. NLP is a computer system that analyze, understand and generate natural human-languages. The input might be text, spoken language, or keyboard process. The process might be translate from language to another, or to understand and represent the content of text in different ways, to build a database represent specific knowledge or generate summaries from long text. Natural language communication with computers has long been a major goal of artificial intelligence, there are many applications of natural language processing developed over the years. They can be mainly divided into two parts as follows:

- **Text-based applications**

  This involves applications such as searching for a certain topic or a keyword in a data base (search engine), extracting information from a large document (information retrieval) that can be discussed later in section 2.3 and translating one language to another (Machine Translation) or summarizing text for different purposes (text summarization) which will be described more in section 2.4.

- **Dialogue based applications**

  Some of the typical examples of this are answering systems that can answer questions, services that can be provided over a telephone without an operator, teaching systems, voice controlled machines (that take instructions by speech) and general problem solving systems.

2.3 Information Retrieval (IR)

IR is the process of obtaining information related to particular topic needed from a collection of information resources, the process of search rely on metadata or on full-text (or other content-based) indexing.

IR work starts when a user writes a query into the system. Queries are formal statements of information needs, for example search strings in web search engines to return information about specific things. The information returned by queries known by object. An object is an entity that is represented by information in a database. The query does not uniquely identify a single object in the collection, but, several objects returned from many resources may be match the query, perhaps with different degrees of relevancy.

User queries are matched against the database information. Depending on the application the data objects may be represented, for example, text documents, images, audio…etc.

Using of Automated Information Retrieval systems (AIR) the returned information are served to reduce what has been called "information overload". IR systems help many universities and public libraries to provide access to books, journals and other documents. Web search engines are the most visible IR applications.

2.4 Text Summarization (TS)

When the problem of "information overload" has grown, the necessity of TS methods are increased. **TS** is the automated process of reducing a text document with the aim of summary that retains the most significant portion of the information in the original document.

There are many of definitions of what text summarization actually means. They include (Josef Steinberger and Karel Je ˇ zek).

- A brief but accurate representation of the contents of a document.
- A distilling of the most important information from a source to produce brief version for a particular user/users and task/tasks'.

The quantitative features which can characterize the summary include:

- Semantic informativeness (can be viewed as a measure of the ability to reconstruct from the summary the original text).
- Coherence.
- Compression ratio.

2.4.1 Types of Text Summarization

The types of summaries include:

Extractive/non-extractive: Extractive type is most used by summarizers is an extractive, choosing portions of the input documents (e.g., sentences) that are believed to be more salient, while non-extractive summarization includes dynamic reformulation of the extracted content, involving a deeper understanding of the input text, and is therefore limited to small domains.

Query-based/ generic: summaries are produced in reference to a user query (e.g., summarize a document about an international summit focusing only on the issues related to the environment) while generic summaries attempt to identify salient information in text without the context of a query.

Single-document/multi s document: the difference between single- document summarization (SDS) and multi-document summarization ((MDS) is quite clear, however some of the types of problems that occur in MDS are qualitatively different from the ones observed in SDS: e.g., addressing redundancy across information sources and dealing with contradictory and complementary information. No true multilingual summarization systems exist yet, however, cross-lingual approaches have been applied successfully (Kirti Bhatia, Dr. Rajendar Chhillar, 2012).

2.5 Sentence Compression

Sentence compression is the task of compressing a long sentence into a short one. It returns the important contents in grammatical form. Different approaches to sentence compression have been suggested by researchers Knight, K. and Marcu, D. (2001).

Knight and Marcu proposed two new data-driven approaches to the sentence compression problem. Both take as input a sequence of words W = w_1, w_2. . . w_n (one sentence). An Algorithm may drop any subset of these words. The words that remain (order unchanged) form a compression. There are 2n compressions to choose from some are reasonable, most are not. Their first approach develops a probabilistic noisy-channel model for sentence compression. The second approach develops a decision-based, deterministic model.

2.6 Works used Noisy Channel for sentences compression

Many works use noisy channel algorithm to compress sentences and documents. One of these works for a document compression used a hierarchical noisy-channel model of text production. The compression system first automatically derives the syntactic structure of each sentence and the overall discourse structure of the text given as input.

Then the system uses a statistical hierarchical model of text creation in order to drop non-important syntactic and discourse constituents so as to generate coherent, grammatical document compressions of arbitrary length (Hal Daum´e and Daniel Marcu, 2009).

A sentence compression system based on synchronous context-free grammars (SCFG), following the successful noisy-channel approach (Michel Galley et al., 2007). In this work Authors defined a head driven Markovization formulation of SCFG deletion rules, which allows to lexicalize probabilities of constituent deletions. Also a robust approach for tree-to-tree alignment between random document-abstract parallel corpora is used, which lets to train lexicalized models with much more data than previous approaches relying exclusively on only just available document-compression corpora. Finally, this work evaluates different Markovized models, and find that their selected best model is one that exploits head-modifier bilexicalization to accurately distinguish adjuncts from complements, and that produces sentences that were judged more grammatical than those generated by previous work.

A novel sentence reduction system was presented for automatically removing irrelevant phrases from sentences that are extracted from a document for summarization purpose. To decide which phrases in an extracted sentence can be removed; the system uses multiple sources of knowledge, including syntactic knowledge, context information, and statistics computed from a corpus which consists of examples written by human professionals. Reduction can significantly improve the conciseness of automatic summaries (Hongyan, 2000). Another work used noisy channel, proposed an application of two different single-document sentence compression methods to the problem of multi-document summarization. The first: a “parse-and-trim” approach, has been implemented in a system called Trimmer and its extended version called Topiary. The second, an HMM-based approach, has been implemented in a system called HMM Hedge. These systems share the basic premise that a textual summary can be constructed by selecting a subset of words, in order, from the original text, with morphological variation allowed for some Word classes. Trimmer selects subsequences of words using a linguistically-motivated algorithm to trim syntactic constituents from sentences until a desired length has been reached (David Zajic et al., 2007).

Most of previous studies in sentence compression depend on the two algorithm mentioned above (decision, noisy Channel), but this research aims to implement sentence compression using semantic role labeling concept. There are two studies based on this concept proposed recently which are Semantic Role Based Sentence Compression (Fatemeh Pourgholamali and Mohsen Kahani, 2012) and Sentence Compression with Semantic Role Constraints (Katsumasa Yoshikawa, et al, 2012). In this research we used the same concept but in deferent way.

2.7 Introduction to Semantic roles labeling (SRL)

SRLis the process of detecting basic event structures such as: who did what to whom, when and where.

Semantic roles (also known as thematic roles or theta roles) attempt to capture similarities and differences in verb meaning that are reflected in argument expression.

There are some characteristics of the theories of the roles:

i. Completeness: Every argument of every verb is assigned some semantic role or other.

ii. Uniqueness: Every argument of every verb is assigned only one semantic role.

iii. Distinctness: Every argument of every verb is distinguished from the other arguments by the role it is assigned. Two levels can be distinguished: strong distinctness, if Uniqueness also holds, and *weak distinctness,* if it does not. In this last case, each argument is assigned a different set of roles from other arguments of the same verb.

***iv.* Independence**: Each role is given a consistent semantic definition that applies to all verbs and all situations. Thus, role definitions do not depend on the meaning of the particular verb or on the other thematic roles it assigns.

2.7.1 Semantic Roles

Below are some semantic roles that can be assigned for each group of words.

Agent: The 'doer' or instigator of the action denoted by the predicate or Instigator of some action.

Example: **John** killed Harry.

Patient: The 'undergoer‹ of the action or event denoted by the predicate.

Theme: The entity that is moved by the action or event denoted by the predicate or Entity undergoing the effect of some action.

Example: **Mary** fell over.

Experiencer: The living entity that experiences the action or event denoted by the predicate or Entity experiencing some psychological state.

Example: **John** felt happy.

Goal: The location or entity in the direction of which something moves or Entity towards which something moves.

Example: John went **home**) (Radford 1997: 326).

Benefactive: The entity that benefits from the action or event denoted by the predicate.

Source: The location or entity from which something moves.

Instrument: The medium by which the action or event denoted by the predicate is carried out.

Locative: The specification of the place where the action or event denoted by the predicate in situated. (Aarts 1997: 88).

Recipient/Possessor: Entity receiving/ possessing some entity.

Example: John got Mary a present.

2.7.2 Importance of SRL

Although the use of SRL systems in real-world applications has thus far been limited, the opinions is promising for extending this type of analysis to many applications requiring some level of semantic interpretation. SRL represents an excellent framework with which to perform research on computational techniques for acquiring and exploiting semantic relations among the different components of a text.

2.7.3 PropBank project

The PropBank project (Kingsbury and Palmer, 2002; Palmer et al., 2005), which provides a large human-annotated corpus of verb predicates and their arguments, has enabled researchers to apply machine learning techniques to develop SRL systems (Gildea and Palmer, 2002; Chen and Rambow, 2003; Gildea and Hockenmaier, 2003; Pradhan et al., 2003; Surdeanu et al., 2003; Pradhan et al., 2004; Xue and Palmer, 2004; Koomen et al., 2005). However, most systems heavily rely on the full syntactic parse trees. Therefore, the overall performance of the system is largely determined by the quality of the automatic syntactic parsers of which the state of the art (Collins, 1999; Charniak, 2001) is still far from perfect(VasinPunyakanok et al.,(2008).

PropBank annotates verb argument structures on top of the syntactic trees of the Penn TreeBank (Marcus et al., 1994). It uses a set of numbered arguments2 (**ARG0, ARG1, ARG2**, etc.) and modifiers (**AM-TMP, AM-MNR**,) etc.). Numbered arguments do not share a common meaning across verbs, they are defined on a notated (Table 2.1). ARG0 and ARG1 are present in most verb-argument structures, other numbered arguments are often not defined in the corresponding frameset and are thus not annotated. Examining PropBank one can also conclude that information regarding **TIME, LOCATION, MANNER, CAUSE and PURPOSE** for a given verb is often present, yet not annotated because the text encoding this knowledge is not a direct syntactic argument of the verb (Eduardo Blanco and Dan Moldovan, 2014).

Table 2.1: Argument modifiers in PropBank

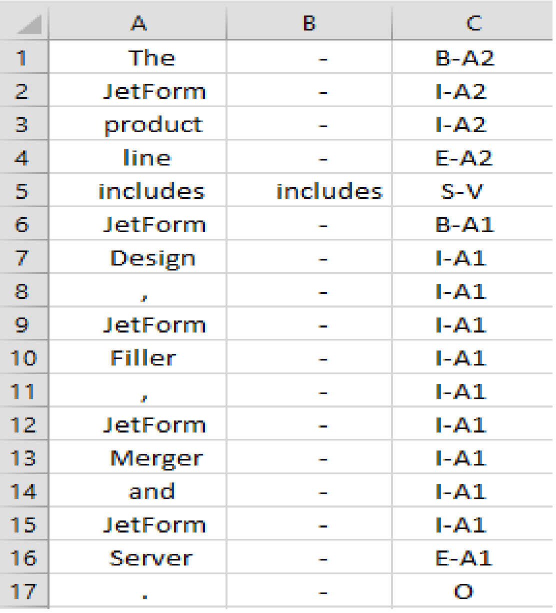
AM-LOC:	location	AM-CAU:	cause
AM-EXT:	extent	AM-TMP:	time
AM-DIS:	discourse connective	AM-PNC:	purpose
AM-ADV:	general-purpose	AM-MNR:	manner
AM-NEG:	negation marker	AM-DIR:	direction
AM-MOD:	modal verb		

2.7.4 Role Extraction Tools

There are many tools to extract roles, some of them direct in websites, like (http://cogcomp.cs.illinois.edu/page/demo_view/14), and other we can install them, like SENNA and SwiRL…etc. In this research, we used SENNA.

SwiRL is a Semantic Role Labeling (SRL) system for English constructed on top of full syntactic analysis of text. The syntactic analysis is performed using Eugene Charniak's parser (included in this package). SwiRL trains one classifier for each argument label using a rich set of syntactic and semantic features. SwiRL is fairly robust, it can work with case-sensitive and case-insensitive text.

Example to extract SRL in the following sentence using SENNA extractor.



	A	B	C
1	The	-	B-A2
2	JetForm	-	I-A2
3	product	-	I-A2
4	line	-	E-A2
5	includes	includes	S-V
6	JetForm	-	B-A1
7	Design	-	I-A1
8	,	-	I-A1
9	JetForm	-	I-A1
10	Filler	-	I-A1
11	,	-	I-A1
12	JetForm	-	I-A1
13	Merger	-	I-A1
14	and	-	I-A1
15	JetForm	-	I-A1
16	Server	-	E-A1
17	.	-	O

Figure 2.1 The role extractor for sentence

2.7.5 Works used SRL to compress sentences

A new unsupervised sentence compression method using (SRL) proposed by (Fatemeh Pourgholamali and Mohsen Kahani, 2012). Sentences are tagged with Part Of Speech tags and semantic role labels. The proposed method depends on the semantic roles of sentences' parts. Moreover, in the process of compression, other sentences in the context are taken into account. The approach is applied in the context of multi-document summarization. Experiments showed better results than other state of the art approaches, were achieved. Another work presented a new semantic constraints to directly capture the relations between a predicate and its arguments (Katsumasa Yoshikawa et al., 2012), whereas the existing approaches have focused on relatively shallow linguistic properties, such as lexical and syntactic information. These constraints are based on semantic roles and superior to the constraints of syntactic dependencies. Their empirical evaluation on the Written News Compression Corpus (Clarke and Lapata, 2008) demonstrates that their system achieves results comparable to other State-of-the-art techniques.

2.8 Data Sets

In this research we used the Ziff–Davis corpus, a collection of newspaper articles announcing computer products, the data set consist ~~from a set~~ of 1067 sentence pairs. Each pair consists of long sentences and it is short sentence from this long, it made by human in English language.

2.9 Sentence Similarity Measures

This research follows sentences similarity measures to test the outputs of our system comparing with human short sentence. Sentences similarity measures , which have wide impact in many text application, such as information retrieval, which is used to assign a ranking score between a query and texts in a corpus. Question answering application requires similarity identification between a question-answer or question-question pair. Furthermore, graph-based summarization also relies on similarity measures in its edge weighting mechanism. Although the gab the variability of natural language expression makes it difficult to determine semantically equivalent sentences. There are many applications have employed certain similarity functions to evaluate sentence similarity, most approaches only compare sentences based on their surface form. As a result, they fail to recognize equivalent sentences at the semantic level. Another issue related to the notions of similarity underlying sentence judgment .Since sentences convey more specific information than documents, a general notion of topicality employed in document similarity might not be appropriate for this task. As Murdoc and Metzler et al point out, there are multiple categories of sentence similarity based on topical specificity. Furthermore, specific notions such as paraphrase or entailment might be needed for certain applications (Pilsen, 2012). The number of similarity measures are so many. In this research we used Jaccard Similarity measures in order to compute the similarity between our proposed method's results and human short sentences and long sentences.

2.9.1 Jaccard Similarity Coefficient

Jaccard Similarity Coefficient (JSC) is a similarity measure that compares the similarity between two feature sets. When applying to sentence similarity task, it is defined as the size of the intersection of the words in the two sentences compared to the size of the union of the words in the two sentences. An example of sets is document sentences and the Jaccard similarity between sentence S1 and S2 defined as in Equation (2.1).

$$\mathrm{Sim}_{\mathrm{Jaccard}}(S1; S2) = \frac{S1 \cup S2}{S1 \cap S2}$$

Equation 2.1. Jaccard Similarity Coefficient

2.10 Summary

We noted that, most of works related to our problem depend on lexical analysis. Our assumption is that, reducing long sentence based on lexical analysis often in consist, non-cohere and diagrammatical research sentence. The literature review showed that researches proposed to improve reduction performance using semantic information are not much. Therefore, this research aims to investigate the use of semantic information (semantic role labeling) in such problem. Chapter 3 will present the general framework of our proposed work, while chapter 4 gives in details how it works with its results and discussion.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter discusses the design of our proposed method to compression sentence using semantic role labeling.

3.2 Methodology:

Our proposed method works as follows:

1. Semantic role labels extraction

In this phase, an input document is directed into SRL extractor in order to extract semantic role for each words.

2. Semantic Role labels Weighing

In this phase, the outputs of phase 1 above are stored in an excel format. This phase aims to study the importance of the roles in both main sentence and human compressed sentence. This importance can be calculated through computing the number of role occurrences overall selected dataset.

In this stage, the human behavior can be investigated by comparing which roles he/she always would like to keep. These roles are expected to carry the important contents in the sentence.

Why this stage is so important?

We designed this stage in order to generate many string of roles, called patterns, to test them since examining natural language processing application performance is very complex issue. Generating more than one pattern gives a good space to analysis the proposed method performance.

3. Sentence Compression

In this phase, we will examine our selected dataset using three patterns. For each pattern, an input sentence will be directed to generate the corresponding short sentence. For each input long sentence we will obtain 3 short sentences using our three designed patterns.

4. Evaluation

For each new generated short sentence (based on the selected pattern) an evaluation with two reference methods will take place. Below is an explanation of what reference methods are.

Normally, to evaluate the performance of NLP or IR methods researchers need to have a reference method (benchmark) to reflect how well or bad the proposed method is. In our case since a work presented by (knight & Marco, 200) is considered the cornerstone, we faced difficulties to implement both methods presented by them (as described them in chapter 2). To implement the case in accepted way, and to let our results measurable, we thought to compare them against two measures:

1. Long sentence vs. human short sentence
2. Long sentence vs. system short sentence
3. Human short sentence vs. system short sentence

These measures try to find out how the performance of our proposed method (system short sentence) is successful compared with human performance (measure 1), how much (the ratio) it removes phrases/words from the original sentence (measure 2) and finally how much our generated results are similar to human performance (measure 3).

Figure 3.1 describes the general framework we proposed. Whereas figure 3.2 shows the same framework but in details. To this end, this chapter ends with describing our proposed method. In next chapter we are going to show implementation steps with attached discussion.

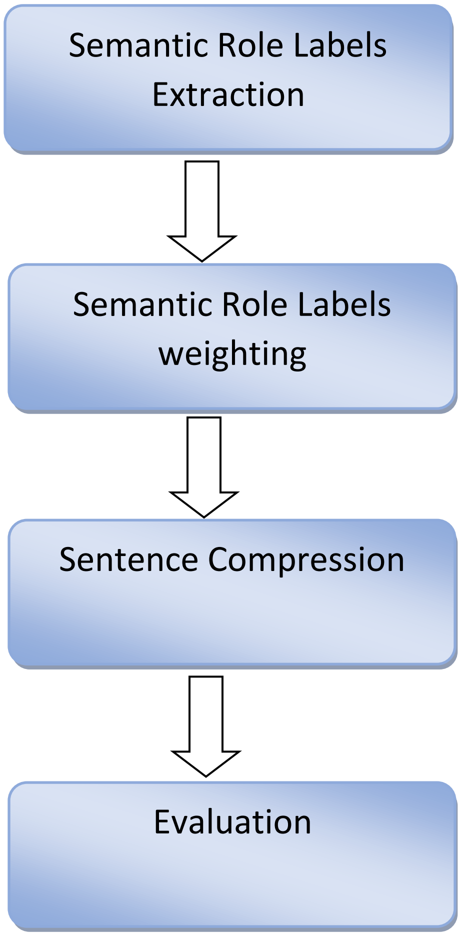

Semantic Role Labels Extraction
Semantic Role Labels weighting
Sentence Compression
Evaluation

Figure 3.1: the general framework of our proposed method

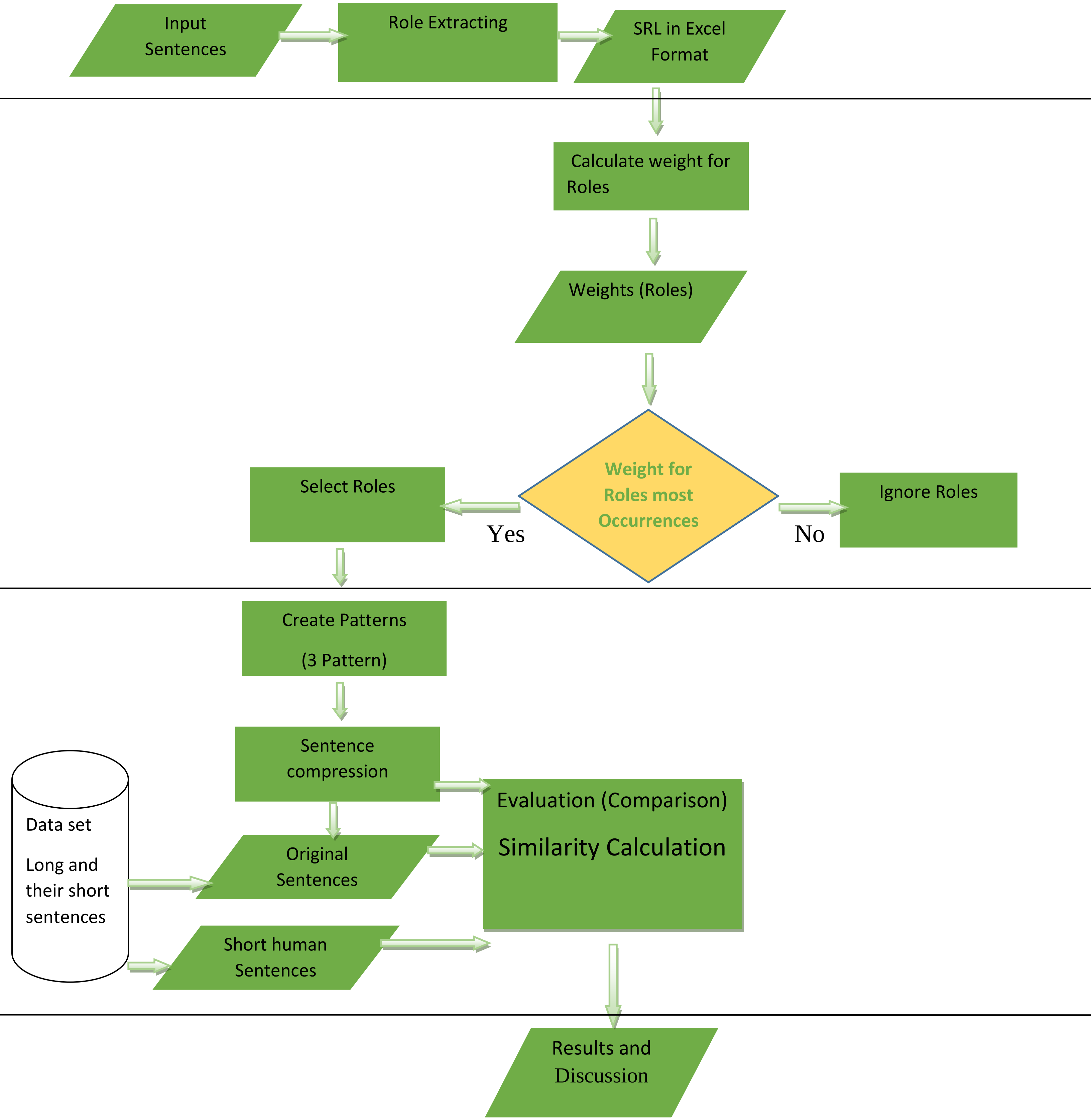


Figure 3.1: the all steps of methodology to compression sentences

CHAPTER 4

4.1 Introduction

This chapter describes step-by-step how our proposed method (4 phrases) is working.

4.1.1 Phase 1: Role Extraction

In Chapter 2 we illustrated the PropBank as a project annotates verb argument structures on top of the syntactic trees of the Penn TreeBank (Marcus et al., 1994). It uses a set of numbered arguments (ARG0, ARG1, ARG2, etc.) and modifiers (AM-TMP, AM-MNR,), then to execute this PropBank project, we used SENNA Role Extractor.

SENNA is a software distributed under a non-commercial license, which outputs a host of Natural Language Processing (NLP) predictions: part-of-speech (POS) tags, chunking (CHK), name entity recognition (NER) and semantic role labeling (SRL). SENNA is fast because it uses a simple architecture, self-contained because it does not rely on the output of existing NLP system, and accurate because it offers state-of-the-art or near state-of-the-art performance. It is written in ANSI C, with about 2500 lines of code. It requires about 150MB of RAM and should run on any IEEE floating point computer.

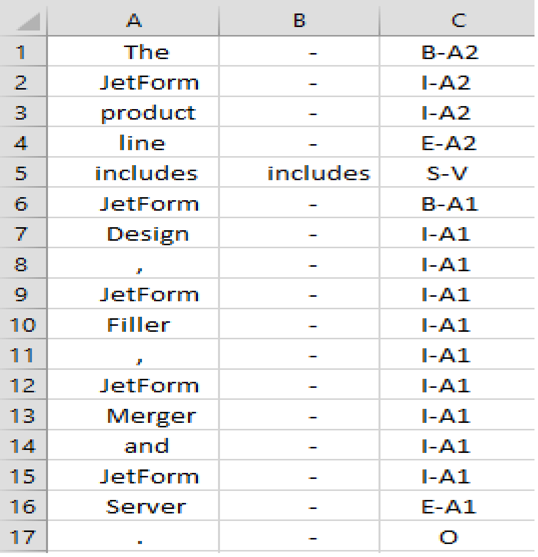
It can running under windows and Linux, It work at command line operating system.

Below. Are two sentences obtained to illustrate the outputs of the SRL Extractor. In addition, Table 4.1 explains the meaning of each roles extracted.

Sentence one:

“The JetForm product line includes JetForm Design , JetForm Filler , JetForm Merger and JetForm Server.”

Figure 4.1 shows the role extraction of the sentence above.



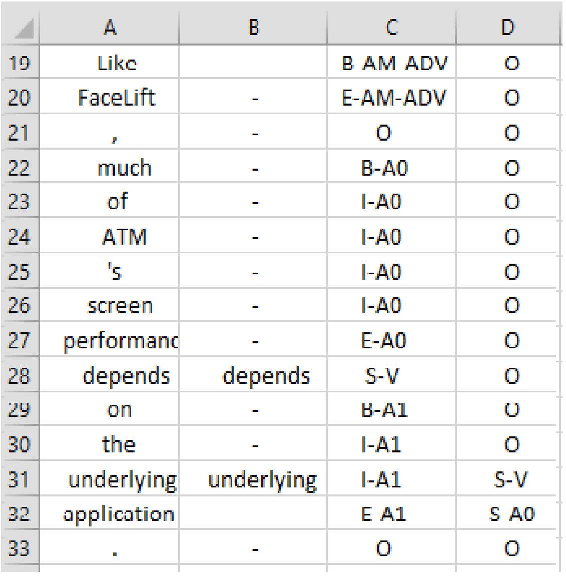
	A	B	C
1	The	-	B-A2
2	JetForm	-	I-A2
3	product	-	I-A2
4	line	-	E-A2
5	includes	includes	S-V
6	JetForm	-	B-A1
7	Design	-	I-A1
8	,	-	I-A1
9	JetForm	-	I-A1
10	Filler	-	I-A1
11	,	-	I-A1
12	JetForm	-	I-A1
13	Merger	-	I-A1
14	and	-	I-A1
15	JetForm	-	I-A1
16	Server	-	E-A1
17	.	-	O

Figure 4.1: role extraction for sentence one in excel file.

Sentence two:

“Like FaceLift , much of ATM 's screen performance depends on the underlying application.”

Figure 4.2 shows the role extraction of the sentence above.



	A	B	C	D
19	Like		B AM ADV	O
20	FaceLift	-	E-AM-ADV	O
21	,	-	O	O
22	much	-	B-A0	O
23	of	-	I-A0	O
24	ATM	-	I-A0	O
25	's	-	I-A0	O
26	screen	-	I-A0	O
27	performanc	-	E-A0	O
28	depends	depends	S-V	O
29	on	-	B-A1	O
30	the	-	I-A1	O
31	underlying	underlying	I-A1	S-V
32	application		E A1	S A0
33	.	-	O	O

Figure 4.2: role extraction for sentence two in excel file.

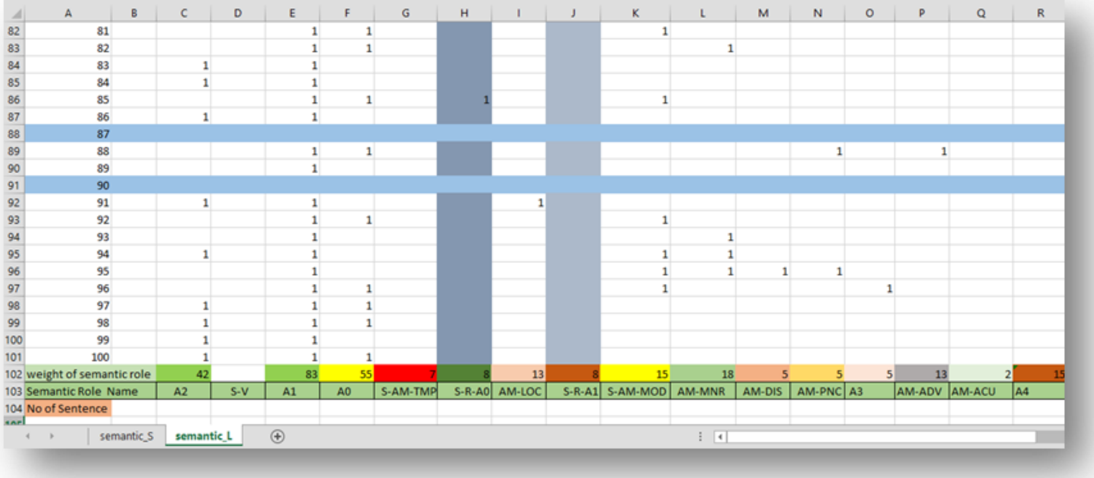
Table 4.1 below explains the meaning of some roles.

Table 4.1: meaning of some SRLs

the role	Interpretation	the role	Interpretation
A0	Causer, Agent, Donor, Assigner	**LOC**	location
A1	Theme, Cognizer, Perceiver, Affected, Possessor, Communicator	**TMP**	time
V	verb	**A2**	Content, Perceived, Possessed

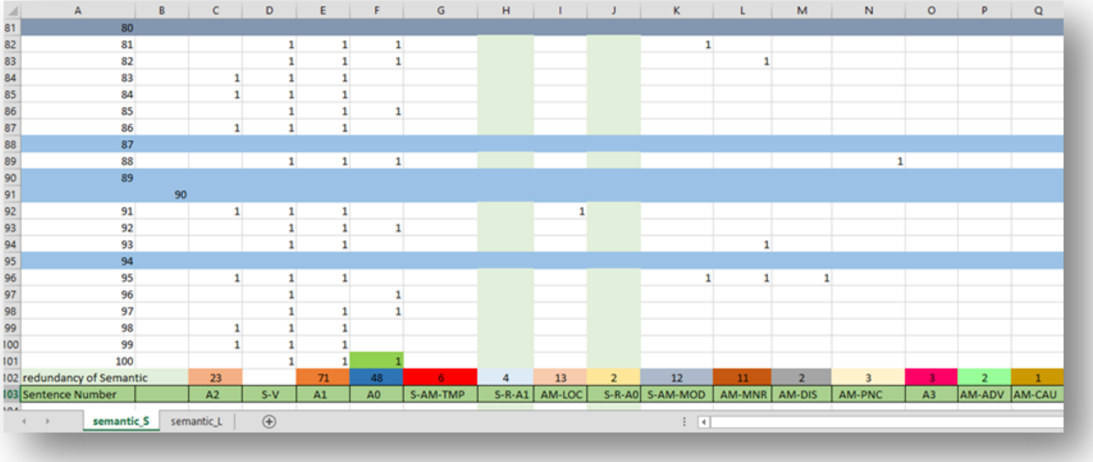
4.1.2 Phase 2: Calculate the weights:

Role weight calculation is computed as follows. An excel file is used as shown in Figure 4.3. The first row represents the names of all 16 roles whereas the columns refer to the sentences. If the role appeared in the sentence, then the corresponding role cell will take value of “1” to indicate role occurrence. If empty then it means no occurrence for the specific role. In Figure 4.4 sentence number 81 is only has 3 roles which are A1, A0, and S-AM-MOD.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
82	81				1	1					1							
83	82				1	1						1						
84	83		1		1													
85	84		1		1													
86	85				1	1		1			1							
87	86		1		1													
88	87																	
89	88				1	1								1		1		
90	89				1													
91	90																	
92	91		1		1				1									
93	92				1	1					1							
94	93				1							1						
95	94		1		1						1	1						
96	95				1						1	1	1	1				
97	96				1	1					1				1			
98	97		1		1	1												
99	98		1		1	1												
100	99		1		1													
101	100		1		1	1												
102	weight of semantic role		42		83	55	7	8	13	8	15	18	5	5	5	13	2	15
103	Semantic Role Name		A2	S-V	A1	A0	S-AM-TMP	S-R-A0	AM-LOC	S-R-A1	S-AM-MOD	AM-MNR	AM-DIS	AM-PNC	A3	AM-ADV	AM-ACU	A4
104	No of Sentence																	

Figure 4.3: weight of roles of original sentences



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
81	80																
82	81			1	1	1					1						
83	82			1	1	1						1					
84	83		1	1	1												
85	84		1	1	1												
86	85			1	1	1											
87	86		1	1	1												
88	87																
89	88			1	1	1								1			
90	89																
91		90															
92	91		1	1	1				1								
93	92			1	1	1											
94	93			1	1							1					
95	94																
96	95		1	1	1						1	1	1				
97	96			1		1											
98	97			1	1	1											
99	98		1	1	1												
100	99		1	1	1												
101	100			1	1	1											
102	redundancy of Semantic		23		71	48	6	4	13	2	12	11	2	3	3	2	1
103	Sentence Number		A2	S-V	A1	A0	S-AM-TMP	S-R-A1	AM-LOC	S-R-A0	S-AM-MOD	AM-MNR	AM-DIS	AM-PNC	A3	AM-ADV	AM-CAU

Figure 4.4: weight of roles for short sentences

Then to compute the weight we used the following equation:

$$\textit{Role-weight} = \frac{\textit{number of role occurrences overall sentences}}{\textit{number of all sentences}}$$

Where “number of all sentences” equals 58. Based on this equation, table 4.2 shows the obtained weight for all roles extracted from original sentences. Whereas Table 4.3 shows the obtained weights for all roles extracted from the human short sentences.

Table 4.2: the obtained weight for all roles extracted from original sentences.

The role	Weight
A0	55
A1	83
A2	42
A3	5
A4	15
V	100
S-AM-TMP	7
S-R-A0	8
S-R-A1	8
AM-LOC	13

AM-MOD	15
AM-MNR	18
AM-DIS	5
AM-PNC	5
AM-ADV	13
AM-ACU	2

Table 4.3: the obtained weights for all roles extracted from the human short sentences.

The role	Weight
A0	48
A1	71
A2	23
A3	3
A4	-
V	100
S-AM-TMP	6
S-R-A0	2
S-R-A1	4
AM-LOC	13

AM-MOD	12
AM-MNR	11
AM-DIS	2
AM-PNC	3
AM-ADV	2
AM-ACU	1

4.1.3 Phase 3: Sentence compression

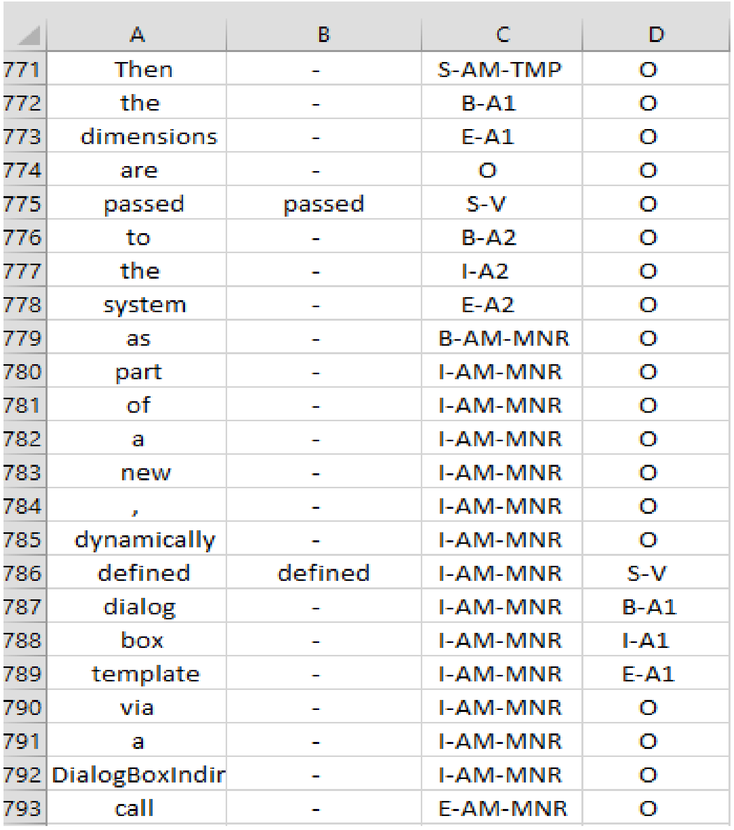
Based on the outputs designed in Phase 2 (Tables 4.2 and 4.3), we proposed to design what so called patterns. The main goal of the “pattern” is to give chance studying how much compressing input sentences is successful. So each sentence will be compressed in three ways then we mark/analyze the performance of each pattern separately. The high similar pattern outputs to our determined reference methods, the more one to be selected for compression rule. **Table 4.4** shows the proposed patterns.

Table 4.4: The contents of all proposed patterns

Pattern Number	Pattern contents
1	A0 V A1 AM-LOC
2	A0 V A1 A2 AM-LOC
3	A0 V A1 A2 AM-LOC AM-DIS AM-TMP

The following sentence is used to describe how others can use our designed patterns for compression. Figure 4.5 below shows the roles extracted for the same sentence.

"Then the dimensions are passed to the system as part of a new, dynamically defined dialog box template via a DialogBoxIndirect call."



	A	B	C	D
771	Then	-	S-AM-TMP	O
772	the	-	B-A1	O
773	dimensions	-	E-A1	O
774	are	-	O	O
775	passed	passed	S-V	O
776	to	-	B-A2	O
777	the	-	I-A2	O
778	system	-	E-A2	O
779	as	-	B-AM-MNR	O
780	part	-	I-AM-MNR	O
781	of	-	I-AM-MNR	O
782	a	-	I-AM-MNR	O
783	new	-	I-AM-MNR	O
784	,	-	I-AM-MNR	O
785	dynamically	-	I-AM-MNR	O
786	defined	defined	I-AM-MNR	S-V
787	dialog	-	I-AM-MNR	B-A1
788	box	-	I-AM-MNR	I-A1
789	template	-	I-AM-MNR	E-A1
790	via	-	I-AM-MNR	O
791	a	-	I-AM-MNR	O
792	DialogBoxIndir	-	I-AM-MNR	O
793	call	-	E-AM-MNR	O

Figure 4.5: represents the role extractor to one sentence

When we use Pattern 1, we will obtain the following reduced output:

"The dimensions are passed."

Pattern 2 consists of all roles in pattern1 in addition to the Role "*A2*". When we use pattern 1, we will obtain the following reduced output:

"The dimensions are passed to the system."

Pattern 3 consists of all roles found in Pattern 2 in addition to the roles "*AM-TMP*" and "*DIS*". Although those two roles are less appeared in the sentences, but they were found at human short sentence. When we apply Pattern 3 on the same sentence, we will obtain the following reduced output:

"Then the dimensions are passed to the system."

4.2 Sentence compression on multi columns:

As we stated in Chapter 2, SRL extractor look first for the verbs in the sentence. It generates a number of analysis (columns) equal to the number of verbs found in the sentence. At each time, the selected verb is considered the main verb. Second, to increase our method performance, we intended to specify main verb from each sentence manually. A human linguistic expertise had helped us to achieve this purpose.

Figure 4.6 shows the compression for sentence in multi verb.

Mac SE and Plus computers have an addressable graphics array of 512 pixel columns by 342 pixel rows , and standard Mac II color displays have an array of 640 by 480 pixels.

Standard Mac II color displays have an arry of 640 by 480 pixels. Jac=1.0

Figure 4.6: compression sentences depend on main verb.

CHAPTER 5

Results and Discussions

5.1 Description

In previous chapter we described how three patterns were selected. From the results of the system, we counted the similarity between the patterns and the human reduction using “Jaccard” algorithm. Table 5.1 shows this similarity results. It consists of four columns as follows:

Column 1: includes the sequence described.

Column 2: includes similarity counts which is computed between human reduced sentences and pattern1 (A0, A1, V, LOC).

Column 3: includes similarity counts which is computed between human reduced sentences and pattern2 (A0, A1, A2, V, LOC).

Column 4: includes similarity counts which is computed between human reduced sentences and pattern 3(A0, A1, A2, V, LOC, TMP, DIS).

Table 5.1: Similarity scores of all reduced Sentences using Pattern 1, Pattern 2 and Pattern 3 against Human-Short Sentences.

Sentence Number	PATTERN 1	PATTERN2	PATTERN3
	A0,A1,V,LOC	A0,A1,A2,V,LOC	A0,A1,A2,V,LOC,TMP,DIS
1	0.68	0.76	0.76
2	0.78	0.78	0.78
3	0.28	0.28	0.28
4	0.7	0.7	0.71
5	0.5	0.5	0.5
6	0.25	0.24	0.24
7	1.0	1.0	1.0
8	0.51	0.50	0.50
9	0.9	0.9	0.81
10	0.72	0.72	0.72
11	0.15	0.32	0.32
12	0.06	0.06	0.06
13	0.76	0.76	0.76
14	0.75	0.75	0.76
15	0.77	0.77	0.77
16	0.67	0.67	0.67
17	1.0	1.0	1.0
18	0.015	0.07	0.07
19	0.55	1.0	1.0
20	0.31	0.31	0.36
21	0.33	0.30	0.30

22	0.33	0.33	0.33
23	0.15	0.15	0.15
24	0.2	0.2	0.24
25	1.0	1.0	1.0
26	0.82	0.82	0.82
27	0.6	0.6	0.6
28	0.135	0.135	0.135
29	0.30	0.30	0.30
30	0.22	0.22	0.22
31	0.81	0.81	0.73
32	0.18	0.73	0.73
33	0.3	0.25	0.25
34	0.06	0.2	0.2
35	0.69	0.69	0.69
36	0.46	0.87	0.94
37	0.02	0.59	0.59
38	0.39	0.39	0.39
39	0.2	0.42	0.42
40	0.82	0.82	0.81

41	1.0	1.0	1.0
42	0.06	0.06	0.06
43	0.34	0.34	0.34
44	0.08	0.08	0.09
45	0.47	0.68	1.0
46	0.46	0.46	0.46
47	0.77	0.77	0.77
48	0.6	0.6	0.6
49	0.56	0.56	0.56
50	0.46	0.81	0.81
51	0.34	0.65	0.65
52	0.29	0.49	0.49
53	1.0	1.0	1.0
54	0.4	0.4	0.4
55	0.13	0.72	0.72
56	1.0	1.0	1.0
57	0.69	0.71	0.71
58	0.22	0.69	0.69

Table 5.1 shows similarity counts between human reduced sentences and sentences reduced using our three proposed patterns. According to Table 4.4, it is clear that when changing between patterns we obtained this score. This refer to adding significant roles based on their weight calculation .For example, compressing sentences with conserving roles “TMP” and “DIS” guided to get sentences with good scores.

Table 5.2 Numbers of rate increment in pattern than others.

Pattern number	Pattern1	Pattern2	Pattern3
Number of Sentences	1	14	20

To simplify the results (scores) found in Table 5.1, we summarized these findings in Tables 5.2 and 5.3. Table 5.2 shows the performance indicator of each pattern. For example, how many sentences Pattern 1 could outperform Patterns 2 & 3? In this case, number of sentences that are obtained high scores compared to other two patterns is only 1 sentence. Whereas number of sentences obtained high score using Pattern 2 are 14 sentences compared to Pattern 1 and Pattern 3. Pattern 3 is mostly the optimal choice to be selected for sentence reduction since it obtained 20 sentences with high scores compared with the first two patterns.

Table 5.3 shows the total average similarity of all patterns scored at Table 5.1. It can be observed that, Pattern3 scores a high average similarity; and for this reason we will depend it for reducing sentences as found in Table 5.4. The second column at Table 5.3 refers to the number of sentences of each pattern when each pattern obtained full similarity (grade 1.00) compared with Human-Short sentence. Again, the number of sentences with score 1.0 using pattern 3 is the highest.

Table 5.3: average similarity for patterns and Similarity Rate's when it equals one.

Pattern Name	Pattern with Human	Number of similarity =1
P1	0.486	6
P2	0.567	7
P3	0.573	8

For all problems presented in chapter3 to evaluate our system, we created another table, it includes the result of 1) The similarity between the original sentences and human reduced sentences (O vs S), and 2) the similarity between the long sentences and pattern3as shown in Table 5.4. In this table we showed the comparison only using for pattern3.

To this end, we showed and a proved how and why we selected pattern 3 to reduce sentences. Next and in Table 5.4 below, we reached to our target/goal experiment to test our proposed method.

To measure the performance of our selected pattern, we computed the similarity scores of both the human and pattern against the original/long sentences. Although this comparison may not be so fair, but as we stated before we should compare our method with human performance. And we justified that we were unable to find a benchmark method to compare with. Therefore, and using Table 5.4, we summarized it in Table 5.5.

Table 5.4: compare the performance result of both Pattern3 and Human against Original Sentences.

Sentence number	Original vs Human (O vs H)	**Original vs system (O vs S)**
1	0.76	1.0
2	0.62	0.78
3	0.19	0.65
4	0.51	0.62
5	0.5	1.0
6	0.27	0.64
7	0.46	0.46
8	0.47	0.91
9	0.78	0.95
10	0.8	0.96
11	0.91	0.31
12	0.4	0.57
13	0.27	0.36
14	0.39	0.5
15	0.14	0.17
16	0.67	1.0
17	0.45	0.45
18	0.1	0.9
19	0.6	0.6
20	0.40	1.0

21	0.67	0.76
22	0.24	0.89
23	0.32	0.81
24	0.42	0.54
25	0.47	0.47
26	0.54	0.51
27	0.6	0.57
28	0.47	0.2
29	0.42	0.35
30	0.54	0.12
31	0.73	1.0
32	0.47	0.52
33	0.43	0.32
34	0.75	0.36
35	0.85	0.71
36	0.74	0.76
37	0.59	1.0
38	0.61	0.23
39	0.82	0.55
40	0.89	0.95
41	0.19	0.19
42	0.24	0.19
43	0.56	0.8
44	0.36	0.31
45	1.0	1.0
46	0.48	0.84
47	0.77	1.0
48	0.67	0.51
49	0.61	0.61

50	0.81	1.0
51	0.25	0.16
52	0.88	0.54
53	0.81	0.81
54	0.77	0.6
55	0.72	1.0
56	0.81	0.81
57	0.71	1.0
58	0.69	1.0

From Table 5.4 we calculate the counts of similarity for Human and our system against original, then described the result in Table 5.5, from which we found that pattern3 has highest similarity than Human in 32 sentences, this illustrates that our system investigate the goal of maintaining the most important information in the sentences, since it is near to original sentence than human sentences. Also Table 5.5 shows that our system vs original equals the human vs original in 9 sentence from 58 sentences. Which indicates our system is accepted.

Table 5.5: illustrates the evaluation of result of our system**.**

	Pattern 3> Human	**Pattern 3 = Human**	**Pattern 3 < Human**
Number of Sentences	**32**	**9**	**16**
Average Similarity	**0.55**	**0.15**	**0.27**

5.2 CONCLUSION

We started this chapter by establishing an experiment which involves as shown similarity calculator of all proposed patterns as shown in Table 5.1. In Table 5.2, we summarized the performance of each pattern; for instance, it is clear that pattern 3 has obtained the highest number of sentences in term of similarity with long sentences. Then we counted the average similarity for all patterns and their number of sentences that scored full similarity (=1). From these two tables, we justified why we selected pattern3 for sequenced experiment. Last, in Table 5.5 we counted the number of sentences under 3 cases. And a result showed that scores of the number of sentences that pattern 3 has outperformed the human was 32 in total of 58 sentences. Thus, we considered this as an accepted result and performance indication.

To this end we concluded to that, the use of pattern 3 as a sentence compressor model is approved and can be used. In addition, additional research can be proposed for developing it.

References

Chandrasekar, R. and Doran,C. and Srinivas.B,.”Motivations and Methods for Text Simplification”.

Josef Steinberger and Karel Je ˇzek, “Text Summarization: An Old Challenge and New Approaches”, 2007.

Kirti Bhatia, Dr. Rajendar Chhillar, “A Statistical Approach to perform Web Based Summarization”, IOSR Journal of Computer Engineering (IOSRJCE) 2278-0661, 2012.

Knight, K. and Marcu, D., “Summarization beyond sentence extraction: A probabilistic approach to sentence compression”, 2001.

Hal Daum´e III and Daniel Marcu “a Noisy-Channel Model for Document Compression”, 2009.

Michel Galley and Kathleen R. McKeown,” Lexicalized Markov Grammars for Sentence Compression”, 2007.

Hongyan,” Sentence Reduction for Automatic Text Summarization”, 2000.

David Zajic, Bonnie J. Dorr, Jimmy Lin, Richard Schwartz Multi-Candidate Reduction: “Sentence Compression as a Tool for Document Summarization Tasks” Journal Information Processing and Management: an International Journal, 2007.

VasinPunyakanok, DanRoth, Wen-tauYih,” The Importance of Syntactic Parsing and Inference in Semantic Role Labeling”, 2008.

Infer Semantic Relations” Association for Computational Linguistics, 2014.

http://www.surdeanu.info/mihai/swirl/ (accessed 16.9.2014).

Fatemeh Pourgholamali and Mohsen Kahani,” Semantic Role Based Sentence Compression” International eConference on Computer and Knowledge Engineering (ICCKE) 2012.

Katsumasa Yoshikawa, Ryu Iida, Tsutomu Hirao, and Manabu Okumura,” Sentence Compression with Semantic Role Constraints” 2007.

Pilsen, “Advanced Methods for Sentence Semantic Similarity “2012.