Formal Description of Arabic Syntactic Structure in the Framework of the Government and Binding Theory

Hammo Bassam[1], Moubaiddin Asma[2], Obeid Nadim[1], and Tuffaha Abeer[1]

[1] KASIT, CIS Department,
[2] Department of Linguistics,
The University of Jordan, Amman,
Jordan

{b.hammo, a.mobaiddin, obein}@ju.edu.jo, atuffaha@hotmail.com

Abstract. The research focus in our paper is twofold: (a) to examine the extent to which simple Arabic sentence structures comply with the Government and Binding Theory (GB), and (b) to implement a simple Arabic Context Free Grammar (CFG) parser to analyze input sentence structures to improve some Arabic Natural Language Processing (ANLP) Applications. Here we present a parser that employs Chomsky's Government and Binding (GB) theory to better understand the syntactic structure of Arabic sentences. We consider different simple word orders in Arabic and show how they are derived. We analyze different sentence orders including Subject-Verb-Object (SVO), Verb-Object-Subject (VOS), Verb-Subject-Object (VSO), nominal sentences, nominal sentences beginning with *inna* (and sisters) and question sentences. We tackle the analysis of the structures to develop syntactic rules for a fragment of Arabic grammar. We include two sets of rules: (1) rules on sentence structures that do not account for case and (2) rules on sentence structures that account for case of Noun Phrases (NPs). We present an implementation of the grammar rules in Prolog. The experiments revealed high accuracy in case assignment in Modern Standard Arabic (MSA) in the light of GB theory especially when the input sentences are tagged with identification of end cases.

Keywords. Arabic syntax, Government and Binding theory, Arabic parser, Arabic natural language processing.

1 Introduction

Words convey meaning. But when they are grouped together based on grammatical structure they convey larger meanings [15]. Identifying the structure (syntax) is the first step towards understanding the meaning of a sentence. Syntactic analysis (parsing) is a procedure that recognizes a sentence and discovers how it is built (i.e., gives its grammatical structure). Recognition involves finding out whether the sentence under consideration belongs to a particular language, i.e., whether it follows all the rules that this language prescribes. Discovering the structure (parsing) involves identifying and marking the various components of a sentence (i.e., phrases and individual parts of speech such as noun, verb, preposition, etc.) [14].

Parsing of sentences is a necessary mechanism for many natural language processing (NLP) applications. Not all NLP applications require a complete syntactic analysis. For information retrieval (IR), it is sufficient to find noun phrases (NPs) and verbal phrases (VPs). However, for such applications as information extraction (IE), text summarization (TS), question answering (QA), we are interested in information about specific syntactic and semantic (meaningful) roles such as agents, objects, locations, time, among others (who did what to whom, when, where, why, etc.).

The key idea in rule-based parsing is that given a grammar and a sentence, a parser will determine if a given sentence is well formed according to the grammar, and what a derivation tree would look like. Despite the fact that the development and maintenance of handwritten grammars is a hard task, there is a strong advantage of rule-based parsing as one can easily modify and accommodate the parser to new tasks. Diab et al. [7], suggested that rule-

ISSN 1405-5546
DOI: 10.13053/CyS-18-3-2017

based parsers are implausible and that syntactic analyzers (parsers) could be based on lexical properties and structure determining principles.

Incorporating the Government and Binding (GB) theory [3], [5] into a parser helps to eliminate many grammar rules because of their redundancy, as syntactic structures can be derivable using means other than explicit rules. Principles are general constraints on syntactic representations (and not on rule application). GB enables linguists to replace many traditional rules using a small number of fundamental linguistic principles. GB principles are constraints over X-bar structures.

The significance of the principles is to constrain the class of possible syntactic representations. The bound on syntactic representation, with language-specific rules, enables a parser to predict syntactic structure(s).

Parsing Arabic texts is challenging because the Arabic language has rich morphology due to its highly inflectional nature, highly flexible word order and frequent use of clitics, which are attached to words [2], [20]. The emphasis in this paper is on the use of Government and Binding (GB) theory in analyzing the syntactic structure of some simple Modern Standard Arabic (MSA) sentences.

We describe a parser that is based on GB grammatical theory. We shall use GB principles and rules to describe Arabic-specific properties or marked structures and to analyze the syntactic structure of some simple Arabic sentences [4], [12]. We consider different word orders in Arabic and show how they are derived. We shall include an analysis of SVO, VOS, VSO, nominal sentences, nominal sentences beginning with *inna* (and sisters), and question sentences. We use this analysis to develop syntactic rules for a fragment of Arabic grammar. Due to space limitation, we shall not present some important computational steps and the structure of a lexicon necessary to build the implemented system.

This paper proceeds as follows: in the next two subsections we present a brief introduction to Arabic and give a brief presentation of the GB theory. In Section 2 we discuss the notion of Arabic syntactic analysis in light of GB. Sections 3 and 4 are dedicated to the parser and its implementation.

1.1 Brief Introduction to Arabic

Arabic is a Semitic language that has a rich morphology and a flexible word order. In this paper we are concerned with Modern Standard Arabic (MSA), which is used in modern writing and is understood by Arabic language speakers. Arabic grammar distinguishes between two types of sentences, verbal and nominal. Nominal sentences have two parts: a subject (*mobtada'* مبتدأ) and a predicate (*khabar* خبر).

When the nominal sentence speaks about being, i.e., if the verb of the sentence is ‘to be’ in English, this verb is not given in Arabic. Arabic morphology is based on roots and patterns through which words are derived. An Arabic word may be composed of a stem consisting of a base root and a pattern which defines its semantic and syntactical role. Moreover, affixes and clitics are often attached to words.

Affixes include inflectional markers for tense, gender, and number. Clitics include prepositions, conjunctions, determiners, and possessive pronouns. Here we present some of the characteristics and / or challenges of the Arabic language.

1. It has a relatively free word order. It is not uncommon to find VSO, SVO and VOS word orders within an Arabic text as in the following examples (see Table 1). All of the sentences in Table 1 are grammatically well-formed and have the same English meaning: “The teacher read the lesson”.
2. Arabic is a clitic or clitic-directed language. Clitics are morphemes that have the syntactic characteristics of a word but are morphologically bound to other words (e.g., coordinating conjunctions, the definite article, many prepositions and particles, and a class of pronouns that attach themselves either to the beginning or the end of words) as in كتبنا :*katabna* (we wrote) which is made up of the verb كتب *katab* and the clitic نا *na* which acts as the subject for the verb *katab* كتب
3. The absence of diacritics (syntactic marks) in most written Arabic texts is very common.

Table 1. Examples of Arabic sentences with different word orders

Order	Arabic example (reads right → left)		
VSO	الدرسَ	المعلمَ	قرأ
	a-dars-a	al-mualim-u	qar'-a
	the lesson	*the teacher*	*read*
SVO	الدرسَ	قرأ	المعلمَ
	a-dars-a	qar'-a	al-mualim-u
	the lesson	*read*	*the teacher*
VOS	المعلمَ	الدرسَ	قرأ
	al-mualim-u	a-dars-a	qar'-a
	the teacher	*the lesson*	*read*

4. Arabic is a pro-drop language. The subject can be omitted, leaving any syntactic parser with the challenge to decide whether there is an omitted pronoun in the subject position.
5. Homographs of words with/without the same pronunciation are often produced. They have different meanings and usually different parts of speech (POS). For example, the Arabic word ذهب :*thahab* can be interpreted as *thahab-a* (a verb meaning "went") or as *thahab-un* (a noun meaning "gold").

1.2 Brief Introduction to the Government and Binding Theory (GB)

The GB theory [3], [5] is an approach to Universal Grammar which includes rules and principles that apply to all languages. However, while certain grammatical principles and rules are universal, there is a lot of variation between different languages such as different ordering for subject (S), verb (V) and object (O). It is agreed that every language has a basic word order, and all other word orders result from the movement of sentence constituents and this movement is restricted by some rules and principles. Words are organized hierarchically into bigger units called phrases. Phrase constituents include:

1. IP – Inflectional Phrase: a phrase headed by I/INFL. I/INFL stands for inflection, and it consists of tense, number, and gender agreement (AGR) elements.
2. CP – Complementizer Phrase: a phrase headed by a complementizer C. C takes an IP(INFL Phrase) as its complement and heads the maximal projection CP.
3. NP – Noun Phrase: a phrase headed by a noun (N).
4. VP – Verb Phrase: a phrase headed by a verb (V).
5. AP – Adjective or Adjectival Phrase: a phrase headed by an adjective (A).
6. PP – Prepositional Phrase: a phrase headed by a preposition (P).

The main principles of GB are:

1. Government, which is concerned with syntactic relations in a sentence and has its main application in case assignment.
2. Theta Theory, which is concerned with describing thematic relations between arguments and predicates.
3. Predicates and arguments: arguments are participants minimally involved in the activity or state expressed by a predicate.
4. Case Theory which is concerned with the assignment of abstract cases (nominative, accusative, and genitive) to words, based on their positions in a sentence.
5. X-Bar Theory, which is concerned with phrase formation. It states that all phrases are headed by a lexical head (noun, verb, adjective, or preposition).
6. Complements combine with X to form X' projections, adjuncts combine with X' to form X''. A specifier combines with the topmost X' to form the maximal projection X''/XP.
7. D-structure and S-structure: all sentences are represented in terms of both forms, the D-structure and the S-structure. The D-structure encodes predicate-argument relations and thematic properties of a sentence and it is built upon the basic word order. The S-structure accounts for the surface ordering of the sentence constituents.
8. NP-Movement: GB assumes that the different word orders arise from the movement of sentence constituents. Hence, a basic word order is assumed, and all other word orders are derived.

Table 2. Rules and examples of Arabic noun phrases (NP)

Syntactic structure	Arabic example	Transliteration	English meaning
NP→ N	كتابٌ	kitab-n	Book
NP→ Det N	الكتابُ	alkitab-u	The book
NP→ NP NP	كتابُ البنتِ	kitab-u el-bint-i	The girl's book
NP→ NP Conj NP	الليلُ والنهارُ	al-layl-u wa anahar-u	The night & day
NP→ NP AP	كتابٌ مفيدٌ	kitab-n mufid-n	A useful book

Table 3. Rules and examples of Arabic adjective phrases (AP)

Syntactic structure	Arabic example	Transliteration	English meaning
AP→ A	مفيدٌ	mufid_n	Useful
AP→ A AP	أزرقٌ داكنٌ	azrak-n dake-n	Dark blue
AP→ AP Conj AP	أزرقٌ وداكنٌ	azrak-n wa dake-n	Dark & blue

Table 4. Rules and examples of Arabic verb phrases (VP)

Syntactic structure	Arabic example	Transliteration	English Meaning
VP→V	قرأ	qar'-a	He read
VP→V NP	أكلَ التفاحةَ	akal-a atufahat-a	He ate an apple
V→V PP	ذهبَ إلى المدرسةِ	thahab-a ila almadrasat-i	He went to school
VP→VP PP	وجدتُ الكتابَ على الطاولةِ	wajadt-u alkitab-a ala atawilat-i	I found the book on the table

Table 5. Rules and examples of Arabic prepositional phrases (PP)

Syntactic structure	Arabic example	Transli-teration	English meaning
PP→ P NP	في المكتبةِ	fi almaktabat-i	In the library

2 Analysis of Arabic Syntax in Light of GB

In this section, we describe our analysis concerning some of the Arabic grammatical structures in the light of the Government and Binding (GB) Theory. We delve into the basics of GB and attempt to apply it on some simple sentence structures in Arabic.

2.1 Scope of Sentence Structure Analysis

Our implementation of Arabic syntactic analysis is restricted to basic sentence structures that include

1. SVO, VOS, and VSO sentences where the subject is an NP and the object is an NP.
2. Sentences followed by a PP adjunct.

3. Nominal sentences made up from NP(s), or NP followed by PP, and nominal sentences preceded by *inna* (and sisters).
4. Question sentences staring with a question word followed by a VSO order sentence.

2.2 Constituent Structure in Arabic

We first analyze the lexical formation of the smaller phrase constituents that make up a sentence. The analysis includes a noun phrase (NP), an adjective phrase (AP), a verbal phrase (VP), and a prepositional phrase (PP).

2.2.1 Noun Phrase (NP)

An NP head is a noun and it can be represented according to the rules and examples presented in Table 2.

2.2.2 Adjective Phrase (AP)

An AP head is an adjective and it can be represented according to the rules and examples presented in Table 3.

2.2.3 Verb Phrase (VP)

A VP head is a verb and can be represented according to the rules and examples presented in Table 4.

2.2.4 Prepositional Phrase (PP)

A PP head is a preposition and it can be represented according to the rules and examples presented in Table 5.

2.3 Arabic Basic Word Order (SVO)

Greenberg [10] claimed that languages which exhibit a Verb-Subject-Object (VSO) word order are a minority among the world languages. If such a claim is valid, then a change in word order is expected to be in the direction of the more common SVO order. Classical Arabic can be considered as one of the VSO languages. According to El-Yasin [9], Colloquial Jordanian Arabic seems to exhibit SVO order judging by the facts of subject-verb agreement and facts about the number of topics allowed to precede sentences in this dialect of Arabic. In [9], the authors concluded that Arabic would be an example of a language changing from a VSO language (in its classical form) into a SVO language (in the case of Jordanian), thus supporting Greenberg's claim [10].

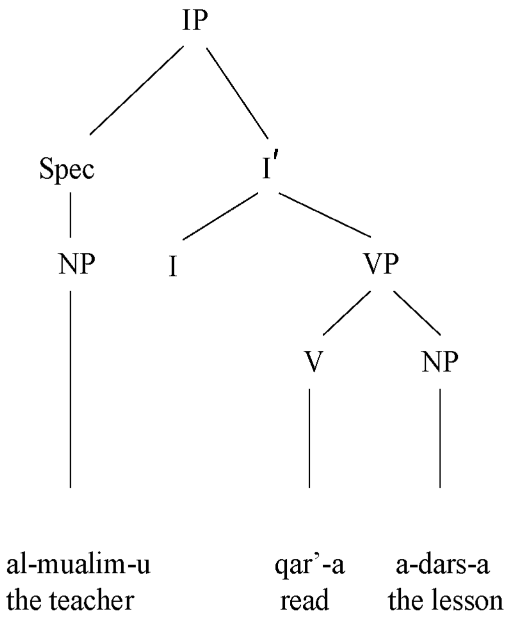

Fig. 1. S-Structure of the SVO Arabic sentence "al-mualim-u qar'-a a-dars-a"

We assume that the basic word order for Arabic sentences within the framework of GB is SVO. In SVO order, I/NFL assigns a NOM case to the subject at [Spec, IP] position (through the percolation of I/NFL to IP), and the verb which heads the VP assigns an ACC case to its object. As an example, consider the following sentence:

المُعلمُ قرأ الدرس (*reads from right to left*) (al-mualim-u qar'-a a-dars-a) "The teacher read the lesson". In this sentence "al-mualim-u" (the teacher), receives a NOM case from I/NFL, and the noun "a-dars-a" (the lesson) is assigned an ACC case from its governing verb "qar'-a" (reads). As a result of I/NFL's government, agreement is imposed. Here, we notice that there is a full agreement in number and gender between the verb and the subject in SVO order as shown in Figure 1.

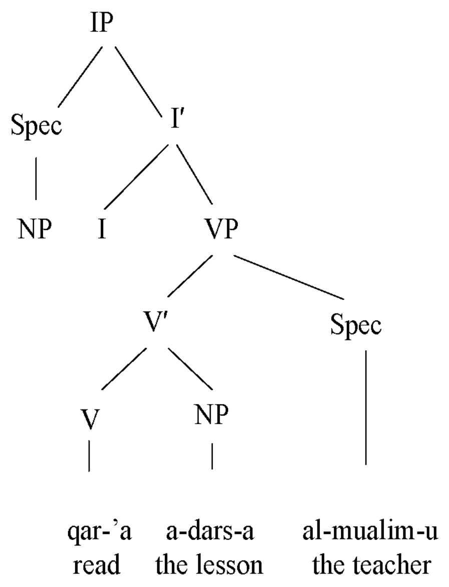

Fig. 2. Structure of the VOS Arabic sentence “qar'-a a-dars-a al-mualim-u”

2.4 Other Word Orders and NP Movement

The other word orders are the result of movements applied on the basic word order. These include the word orders we discuss below.

2.4.1 VOS Word Order

VOS results from the subject adjunction to the end of VP. Hence, it will receive a NOM case from I/NFL. And to satisfy the EPP principle, we can assume [Spec, IP] to be occupied by PRO. Figure 2 explains the VOS word order for Arabic.

2.4.2 VSO Word Order

In the D-structure (Figure 3(1)), the subject at [Spec, IP] receives a NOM case from I/NFL, the verb's object receives an ACC from the verb. VSO order is obtained by moving the verb to empty [C, CP] (Head to Head movement) leaving its co-indexed trace (the accusative case of VP's internal NP is assigned through the verb's trace) [1] (see Figure 3(2)).

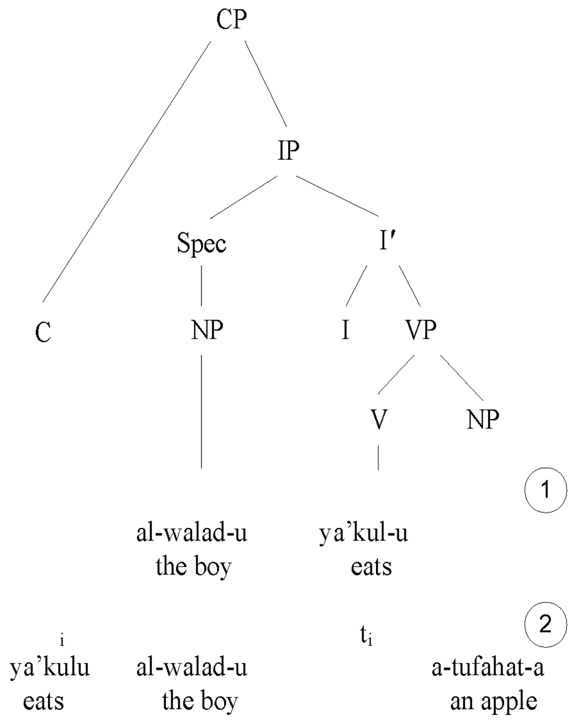

Fig. 3. (1) The D-structure and (2) the S-structure of the VSO Arabic sentence

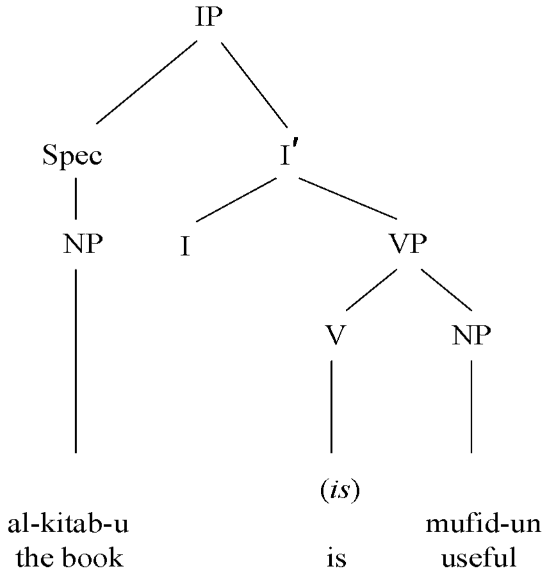

Fig. 4. The structure of a typical Arabic nominal sentence

ISSN 1405-5546
DOI: 10.13053/CyS-18-3-2017

2.5 Nominal Verbless Sentences

2.5.1 Typical Nominal Sentences

A typical Noun Phrase (NP) in Arabic contains two nouns as in the following example: الكتابُ مفيد (al-kitab-u mufid-n) "The book is useful".

As mentioned earlier, we consider SVO to be the basic word order for Arabic sentences. To explain the grammaticality of this type of sentences, we can assume that there is a hidden verb, such that this verb carries the meaning of *is*, and it occupies V. Accordingly, the phrase will be tensed, which allows I/NFL to assign a nominative case to NP at [Spec, IP].

However, the hidden verb will fail to govern its internal argument. To solve this issue we will adopt the default case approach mentioned in [13] which says that NPs with no case assigner are possible in Arabic and they are assigned a nominative case. Figure 4 shows the structure of a typical nominal sentence الكتابُ مفيدٌ (al-kitab-u mufid-n) "the book is useful". "al-kitab-u" receives a NOM case from I/NFL, and "mufid-n" is assigned the default NOM case.

2.5.2 Nominal Sentences with *inna* (and sisters)

Inna and sisters particles (إنّ وأخواتها) can occur in nominal sentences. They include the particles إنّ، أنّ، ليت، لعلّ - inna, anna, layta, laalla... . *Inna* and sisters are complementizers that assign an accusative case to their noun governees.

Consider the following example: إنّ الكتاب مفيد (*inna* al-kitab-a mufid-un) "the book is useful".

In Arabic, in the existence of *inna* and sisters particles at [C, CP], the NP at [Spec ,IP] is assigned an ACC case from the complementizer particle. We assume that *inna* and sisters complementizers are stronger than tensed I/NFL and prevents it from assigning the nominative case to NP at [Spec, IP]. In the above example, *inna* is a complementizer that assigns an ACC case to its governee "al-kitab-u" and the hidden verb (i*s*) fails to assign an accusative case to "mufid-n". Therefore, it is assigned the default NOM case as shown in Figure 5 (1).

Now, consider the sentence إنّ الولد يأكل التفاحة (*inna* al-walad-a ya'kul-u a-tufahat-a) "the boy is eating an apple". In this sentence, the complementizer *inna* assigns an ACC case to "al-walad-a", and the verb "ya'kul-u" assigns an ACC case to its object "al-tufahat-a" (see Figure 5 (2)). The same applies to all other *inna* sisters' complementizers.

2.5.3 Question Sentences

Questions in Arabic usually start with a question word such as من (man) "who", ماذا (matha) "what", متى(mata) "when" and أين(ayna) "where".

– *Questions on Subjects or Objects.* Both من (who) and ماذا (what) can be used to ask about the subject or the object. If a question is about the subject, in the D-structure (Figure 6(1)) the question word is placed at [Spec, IP] and in the S-structure (Figure 6(2)) it is moved to [Spec, CP].

If a question is about the object, in the D-structure the question word is placed at the object's node under VP, and to produce the S-structure, the question word is moved to [Spec, CP]. Consider the following cases:

– *Questions on Subjects*. من قَرأ الدرس؟ (man qar'a a-dars-a?) "Who read the lesson?"

In the D-structure (Figure 6(1)), the base of the question word "man" is generated at [Spec, IP], but in the S-structure (Figure 6(2)), it moves to [Spec, CP].

– *Questions on Objects*. ماذا قرأ الطالب؟ (matha qar'-a a-talib-u?) "What did the student read?"

In the D-structure (see Figure 7), the base of the question word "matha" is generated at the object position. One can notice that this question begins with a question word followed by a verb. In order to explain the sentence grammaticality, we assume that the question word is moved to [Spec, CP], and the verb is moved to [C, CP] (see Figure 8).

– *Questions on VP Adjuncts*. أين (ayn-a) "where", متى(mata) "when", and كيفَ(kayf-a) "how" are usually used to ask about a VP's adjunct. Consider the following example: أينَ سافر علي؟ (ayna safar-a Ali?) "Where did Ali travel?"

In the D-structure, the base of the question word "ayn-a" is generated at VP's adjunct

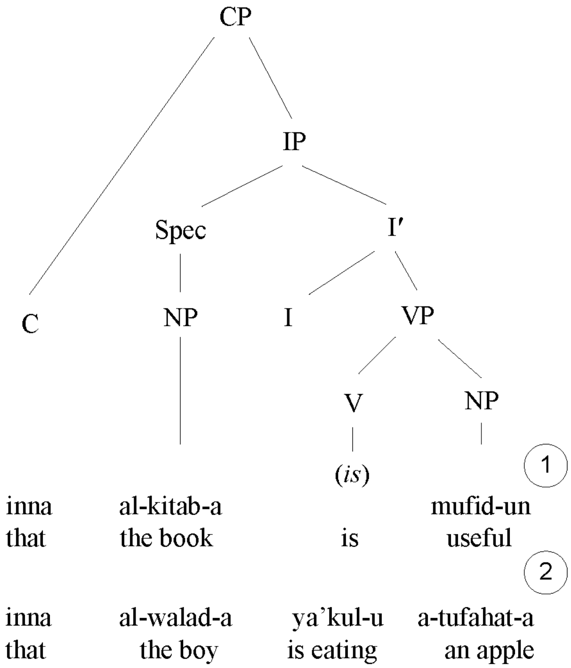

Fig. 5. Two nominal sentences (1) with *inna* and a hidden verb, (2) with *inna* and a verb

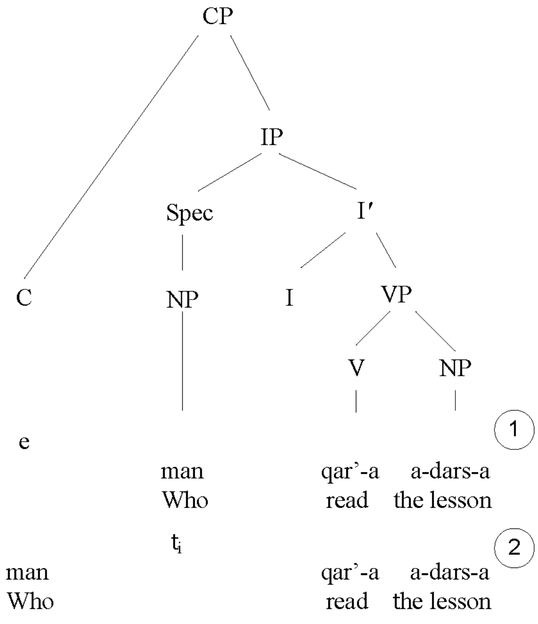

Fig. 6. (1) The D-structure and (2) the S-structure of a question on subject with the question word *man*

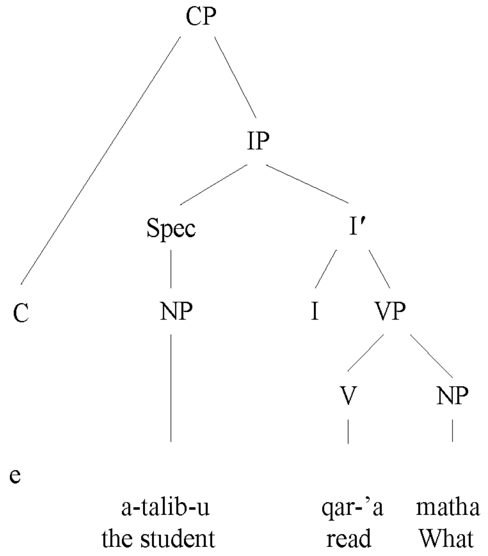

Fig. 7. D-structure of a question on object with the question word *matha*

position, and in the S-structure, we assume that the question word is moved to occupy [Spec,CP], and the verb is moved to [C, CP].

2.5.4 Yes/No Questions

Many Yes/No questions use the question word هل (hal) "Did". Here is an example to explain this case: هل سافرعلي؟ (hal safar-a Ali?) "Did Ali travel?"

In this sentence, in the D-structure, we assume that the base of the question word "hal" is generated as an IP adjunct. To produce the S-structure, the verb is moved to [C, CP] and the question word is moved to [Spec, CP].

3 The Arabic Parser

Our syntactic parser (see Figure 9) takes a sentence as an input and outputs whether it is syntactically correct also generating its bracket structure. To assign part of speech tags (POSTs) to the sentence, we used the Aramorph tagger [4], which is the Java version of Buckwalter's Arabic morphological analyzer.

Unlike rule-based grammars that use a large number of rules to describe patterns in a language, the GB Theory explains these patterns in terms of more fundamental and universal principles [7], [12], [19]. A key issue in building a principle-based parser is how to interpret in a procedural way the principles expressed as grammar rules. Since GB principles are constraints over syntactic structures, one way to implement the principles is as follows.

1. Generate candidate structures of a given sentence that satisfy X-bar theory and sub-categorization frames of the words in the sentence.
2. Filter out structures that violate any one of the principles.
3. The remaining structures are accepted as parse trees of the sentence.

Once the grammar rules are compiled into Prolog, they receive a procedural interpretation, becoming a top-down, left-to-right, recursive-descent parser. In other words, by representing the rules of grammar as axioms in Prolog horn-clause logic, we can use Prolog theorem proving engine as a parser.

3.1 Grammar Rules

The rules in the grammar base include a set of Arabic grammar rules derived from our analysis of Arabic sentences according to GB. They are divided into two parts. The first part includes the syntactic rules that do not account for words case marks (diacritics) (see Tables 6 and 7) and the second part includes rules for case marked sentences (see Tables 6 and 8). In either case, the basic sentence phrase constituent syntax is listed first, followed by the rules for the syntax of analyzed sentences.

4 System Implementation and Results

An input sentence to the parser is represented as a sequence of tags. The syntactic parser takes the sequence of tags of the tagged sentence and returns as output valid syntactic structure(s) of the sentence. We adopted a top-down recursive approach; rules start at the sentence level (S), continue to rules for phrases (intermediate level) and finally to parts of speech (lowest level).

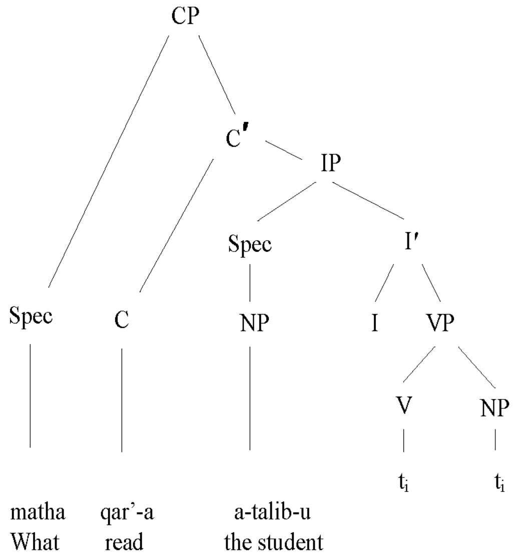

Fig. 8. S-structure of a question on object with the question word *matha*

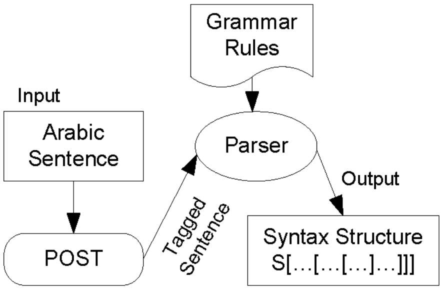

Fig. 9. System architecture

We have opted to employ SICStus Prolog 3.12.2 [21] to implement the parser since Prolog can be effectively used in natural language analysis due to the following: (1) Prolog is a logical programming language, which seems suitable to express grammar rules and (2) we were not aiming at testing the efficacy of the Arabic grammar base. We have employed two files: one is consulted when caseless analysis is required by the user and the other is used when the user considers cases.

Table 6. Arabic grammar rules

Regardless of the case		Regarding the case	
S → CP | IP	AP → Adj	S → CP | IP	AP(Case) → Adj(Case)
S → S Conj S	AP → Adj AP	S → S Conj S	AP(Case) → Adj(Case) AP(Case)
NP → N	AP → AP Conj AP	NP(nom) → N(nom)	AP(Case) → AP(Case) Conj AP(Case)
NP → N NP	PP → Prep NP	NP(acc) → N(acc)	PP → Prep NP(gen)
NP → NP Conj NP	Spec → NP	NP(gen) → N(gen)	Spec(Case) → NP(Case)
NP → NP AP	Spec → Pron	NP(Case) → N(Case) NP(Case2)	Spec(Case) → Pron(Case)
NP → Prop Noun	Spec → Dem	NP(Case) → NP(Case) Conj NP(Case)	Spec(Case) → Dem(Case)
		NP(Case) → NP(Case) AP(Case)	
		NP → Prop Noun	
		Case: a variable specifying NP's case.	
		(Case2 = Case or Case2=gen)	

Table 7. Arabic sentence rules regardless of the case

Order	Rule	Explanation
SVO	IP → Spec VP	
	IP → Pro VP	(Pro: stands for a prodrop pronoun that is hidden)
	VP → V NP	(object is an NP)
	VP → VP PP	(PP adjunct to VP)
VOS	IP → VP	
	VP → VP Spec	
	VP → VP PP	(PP adjunct to VP)
VSO	CP_vso → V IP_vso	
	IP_vso → Spec	(case of intransitive verb)
	IP_vso → Spec VP_vso	
	VP_vso → NP(acc)	
	VP_vso →VP_vso PP	(PP adjunct to VP)
Nominal Sentence	IP_nominal → Spec VP_nominal	
	VP_nominal → NP	(comment is an NP)
	VP_nominal → PP	(comment is a PP)
	CP → Func_word IP_nominal	(*inna* and sisters starting the sentence)
Questions	CP → Q_word CP_vso	

As mentioned earlier, the input to the Prolog parser is a sequence of part of speech tags. The parser matches the input list with sentence rules, breaking the sentence down and matching it with smaller constituent rules until a match is found at the level of tags. If there is a match, the system points out a possible syntax structure upon which the input sentence is built. However, if there is no match, the system returns a "No" result to indicate a mismatch with the sentence rules.

The implemented system was tested on 500 sentences. Sentences were tested against grammar rules for both cased and the caseless sentences. Table 9 shows some sample sentences along with their resulting syntax structures. For all results in the sample, bold syntax structure signifies a correct result. Starting with SVO sentences, Table 9 shows that for all analysis both structures (caseless and cased) are syntactically correct.

For VSO sentences, the cased analysis for the first example is correct. The caseless analysis for the same example produced two structures; the first one is correct while the second one is

ISSN 1405-5546
DOI: 10.13053/CyS-18-3-2017

Table 8. Arabic sentence rules regarding the case

Sentence order	Rule	Explanation
SVO	IP → Spec(nom) VP	
	IP → Pro VP	
	VP → V NP(acc)	(object is an NP)
	VP → VP PP	(PP adjunct to VP)
VOS	IP → VP	
	VP → VP Spec	
	VP → VP PP	(PP adjunct to VP)
VSO	CP_vso → V IP_vso	
	IP_vso → Spec	(when verb is intransitive)
	IP_vso → Spec VP_vso	
	VP_vso → NP(acc)	
	VP_vso →VP_vso PP	(PP adjunct to VP)
Nominal Sentence	IP_nominal → Spec(nom) VP_nominal	
	VP_nominal → NP(nom)	(comment is an NP)
	VP_nominal → PP	(comment is a PP)
Nominal sentence with *inna* and sisters	CP → Func_word IP_nominal2 IP_nominal2→ Spec(acc) VP_nominal	

incorrect. This is because the analyzer could not differentiate between the subject NP and the object NP. The second example shows that both structures are correct.

For VOS sentences, the cased analysis for the first sentence is correct. The caseless analysis for the same sentence produced three structures; the first two are incorrect while the third one is correct. The second example shows a correct analysis of the cased structure while it shows two structures for the caseless syntax, and only the second one is correct.

For the nominal sentences, the caseless and the cased syntaxes were correct. Finally, for the question sentences, in the first example the caseless and the cased analyses produced one structure, which is correct in both cases. The second example produced two structures for the caseless syntax, but it produced only one structure for the cased syntax.

5 Discussion

By now, it is obvious that the implemented syntactic parser based on the GB theory sometimes produces multiple analyses for a sentence. Mainly this is due to semantic ambiguity, which is not expected to be covered by syntactic parsing in the first place. For example, in a sentence which is tagged as [verb, noun, noun], the syntactic parser has no clue about the semantics of the noun words, so it can be interpreted as VSO and VOS as we will explain shortly.

When using caseless analysis, grammatical sentences were parsed correctly and the parser usually produced many possible analyses, especially when the sentence has a verb. This multiplicity in analyses can be explained to a larger extent by the following reasons:

- **Case 1: VSO| VOS| Nominal Sentence**

This case is related to NP ambiguity because the parser has no clue on where a subject and an object start and end.

- **Case 2: VSO| VOS**

In this case the parser is not able to determine whether an NP is a subject or an object.

- **Case 3: SVO| VS**

The parser is not able to determine if there is a hidden subject (PRO) when a sentence has an

SVO order, or if a sentence is VS with an intransitive verb.

When using cased analysis, grammatical sentences were parsed correctly with the highly enhanced precision in determining the exact sentence structure, usually producing one matching syntactic structure for the input sentence. However, the employed Aramorph POST sometimes produced faulty tags, which affected the overall system results and led to wrong analysis of some sentences.

6 Comparison with Previous Work

As far as we know there are very few attempts to develop a GB-based parser for Arabic [13], [16]. Some attempts of Arabic syntactic analysis have been made in [17] [18] [20] based on a lexicon of words which includes their lexical and syntactical features that aid in disambiguating the sentence structure. These approaches employ a morphological analyzer and define rules on the word grammatical categories level: object, subject, etc. A lexicon is divided into three categories: nouns, verbs, and particles. Two types of features are associated with the lexicon entries: syntactic and lexical. Syntactic features are used to resolve syntactic ambiguity such as a verb's tense, subject and object, gender and number. Lexical features are used to resolve lexical ambiguity such as a verb's subject rationality, object rationality and the infinitive form.

Daimi [6] presented an Arabic syntax analyzer where the focus is on finding occurrences of certain types of ambiguous structures in Arabic sentences within a given text. For example, ambiguous situations include the case of 'omitted latent personal pronoun' as in سألزيديوسف أنيذهب :sa'al-a zaid-un yousef-a an yathhab-a where the verb (يذهب) might refer to either (زيد) or (يوسف).

Diab et al. [7] described a statistical model which employs the Support Vector Machine (SVM) approach to identify the base phrase chunks of sentences is used. These authors use a tagged corpus which includes frequencies of tags for each word. The tagged sentences are input to the Base Phrase chunking system which uses trained data to predict phrase classes.

Habash and Rambow [11] presented some work on extracting tree structures for Arabic phrases. They follow a statistics-based approach by using the Penn Arabic Treebank (PATB) to extract a tree grammar for sentences. In their paper the authors stated that they extracted 200 trees. El Hadj et al. [8] presented an approach to part of speech tagging which makes use of sentence structure rules. The authors defined two general finite state machines: one for nominal phrases and the other for verbal phrases.

7 Conclusions and Future Work

Arabic automatic syntactic parsing is an extensive research field due to the richness of the Arabic language. In this paper, we analyzed the syntactic structure of some simple Arabic sentences based on the GB theory. We considered different word orders in Arabic and showed how they were derived. We included an analysis of SVO, VOS, VSO, nominal sentences, nominal sentences with *inna* (and sisters) and question sentences. We have used the analysis to develop syntactic rules for a fragment of Arabic grammar, and we developed two sets of rules: (1) rules on sentence structures that do not account for case and (2) rules on sentence structures that account for NP case.

We presented an implementation of the grammar rules in Prolog. The results showed a high accuracy especially when the input sentences were tagged with identification of end cases. It is important to note that the system is far from complete. We intended to test it on a standard corpus and compare it with similar systems. The proposed system is flexible and can be extended such that further modifications can be applied. We hope to enhance the system by (1) using a morphological analyzer that would provide important features about words such as clitics identification, identification of number and gender features and (2) by adding more rules to deal with more sentence structures and to cover other syntactic features such as subject-verb agreement on number and gender, word clitics and cases that are represented as suffixes to nouns.

Table 9. Examples of caseless and cased syntax for Arabic sentences from our parser (bold syntax structure signifies a correct result)

Sentence	Caseless syntax	Cased syntax
SVO		
الولد يَحب المطالعة al-walad-u yohib-u al-motalaat-a (*The boy likes to read*)	[noun,verb,noun] **[ip: np[vp: verb np]]**	[noun_nom,verb,noun_acc] **[ip:np_nom[vp: verb np_acc]]**
الولد كتب رسالةً إلى صديقه al-walad-u katab-a resalat-n ela sadekeh-i (*The boy wrote a letter to his friend*)	[noun,verb,noun,prep,noun] **[ip: np[vp:[vp: verb np][pp: prep np]]]**	[noun_nom,verb,noun_acc,prep, noun_gen] **[ip: np_nom[vp:[vp: verb np_acc][pp: prep np_gen]]]**
الفتاة أكلت التفاحة الحمراءَ al-fatat-u akalt a-tufahat-a al-hamra-a (*The girl ate the red apple*)	[noun,verb,noun,noun] **[ip: np[vp: verb np]]**	[noun_nom,verb,noun_acc,noun_acc] **[ip: np_nom[vp: verb np_acc]]**
VSO		
ذهبت البنت إلى المدرسة thahabat al-bent-u ela al-madrasat-i (*The girl went to school*)	[verb,noun,prep,noun] **[cp: verb[ip: np[vp:t [pp: prep np]]** [ip:pro [vp:[vp: verb np][pp: prep np]]]	[verb, noun_nom,prep, noun_gen] **[cp: verb[ip: np_nom[vp:t [pp: prep np_gen]]]]**
قرأت الكتاب أشيقاً qaraut-u kitaba-n shaiek-n (*I read an enjoyable book*)	[verb,noun,adj] **[ip:pro [vp: verb np]]**	[verb,noun_acc,adj_acc] **[ip:[vp: verb np_acc]]**
VOS		
قرأ الدرس المعلمَ qara-a a-dars-a almoalem-u (*The teacher read the lesson*)	[verb,noun,noun] [cp: verb[ip: np[vp:t np] [ip:pro [vp: verb np]] **[ip:[vp:[vp: verb np] np]]**	[verb,noun_acc,noun_nom] **[ip:[vp:[vp: verb np_acc] np_nom]]**
جاء إلى البيت زائرَ ja'-a ela al-bayt-i zaer-n (*A visitor came home*)	[verb,prep,noun,noun] [ip:pro [vp: verb[pp: prep np]]] **[ip:[vp:[vp: verb[pp: prep np]] np]]**	[verb,prep,noun_gen,noun_nom] **[ip:[vp:[vp: verb[pp: prep np_gen]] np_nom]]**
Nominal-Sentences		
الجو جميلَ al-jaw-u jamilo-n (*the weather is beautiful*)	[noun,noun] **[ip: np[vp:e np]]**	[noun_nom,noun_nom] **[ip: np_nom[vp:e np_nom]]**
الكتاب َعلى الطاولة al-kitab-u ala a-tawilat-i (*The book is on the table*)	[noun,prep,noun] **[ip:np[pp:e[pp:prep np]]]**	[noun_nom,prep,noun_gen] **[ip:np_nom[pp:e [pp:prep np_gen]]]**
Question-Sentences		
هل سَافر مَحمدٌ؟ Hal safar-a Mohamad-n? (*Did Mohammad leave?*)	[q_word,verb,noun] **[cp: q_word[cp: verb[ip: np]]]**	[q_word,verb,noun_nom] **[cp: q_word[cp: verb[ip: np nom]]]**
من قرأ الدرس؟ man qar'-a a-darsa-a? (*Who read the lesson?)*	[q_word,verb,noun] [cp: q_word[cp: verb[ip: np]]]} **[cp: q_word[ip:t[vp: verb np]]]**	[q_word,verb,noun_acc] **[cp: q_word[ip:t[vp: verb np_acc]]]**

References

1. **Al-Bayaty, J. (1990).** *Adjunction in Arabic, case and chain theory.* Ph.D. thesis, Simon Fraser University, Burnaby, BC, Canada.
2. **Bataineh, B.M. & Bataineh, E.A. (2009).** An efficient recursive transition network parser for Arabic language. *Proceedings of the World Congress on Engineering*, London, UK, 2, 1307–1311.
3. **Black, C.A. (1999).** A step-by-step introduction to the government and binding theory of syntax. *SIL-Mexico Branch and University of North Dakota*. Retrieved from http://www.sil.org/americas/mexico/ling/E002-IntroGB.pdf.
4. **Brihaye, P. (1993).** AraMorph morphological analyzer for Arabic. Retrieved from http://www.nongnu.org/aramorph/
5. **Chomsky, N. (1993).** *Lectures on government and binding: The Pisa lectures (7th ed.).* Berlin; New York: Mouton de Gruyter.
6. **Daimi, K. (2001).** Identifying syntactic ambiguities in single-parse Arabic sentence. *Computers and the Humanities*, 35(3), 333–349.
7. **Diab, M., Hacioglu, K., & Jurafsky, D. (2007).** Automatic processing of modern standard Arabic text. *Arabic Computational Morphology* (159–179). Dordrecht, The Netherlands.
8. **El Hadj, Y.O.M., Al-Sughayeir, I.A., & Al-Ansari, A.M. (2009).** Arabic part-of-speech tagging using the sentence structure. *Proceedings of the Second International Conference on Arabic Language Resources and Tools*, Cairo, Egypt, 241–245.
9. **El-Yasin, M.K. (1985).** Basic word order in classical Arabic and Jordanian Arabic. *Lingua*, 65(1-2), 107–122.
10. **Greenberg, J.H. (1963).** Some universals of grammar with particular reference to the order of meaningful elements. In *J. Greenberg, ed., Universals of Human Language* (73–113). Cambridge, Mass: MIT Press.
11. **Habash, N. & Rambow, O. (2004).** Extracting a tree adjoining grammar from the Penn Arabic Treebank. *JEP-TALN 2004, Session Traitement Automatique de l'Arabe*.
12. **Haegeman, L.M.V. (1991).** *Introduction to government and binding theory*. Oxford, UK; Cambridge, Mass., USA: B. Blackwell.
13. **Homeidi, M.A. (2003).** The notion of governor in modern standard Arabic (MSA) and English: a contrastive perspective. *Journal of King Saud University – Science*, 15, 49–62.
14. **Jurafsky, D. & Martin, J.H. (2000).** *Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition*. Upper Saddle River, N.J.: Prentice Hall.
15. **Lobeck, A. (2000).** *Discovering grammar: an introduction to English sentence structure*. New York: Oxford University Press.
16. **Moubaiddin, A., Tuffaha, A., Hammo, B., & Obeid, N. (2013).** Investigating the syntactic structure of Arabic sentences. *1st International Conference on Communications, Signal Processing, and their Applications (ICCSPA)*, Sharjah, UAE.
17. **Othman, E., Shaalan, K., & Rafea, A. (2004).** Towards resolving ambiguity in understanding Arabic sentences. *International Conference on Arabic Language Resources and Tools, NEMLAR*, Cairo, Egypt. 118–122.
18. **Othman, E., Shaalan, K., & Rafea, A. (2003).** A chart parser for analyzing modern standard Arabic sentence. *Proceedings of the MT Summit IX Workshop on Machine Translation for Semitic Languages,* 37–44.
19. **Riemsdijk, H. & Williams, E. (1986).** *Introduction to the theory of grammar*. Cambridge, Mass.: *MIT Press*.
20. **Shaalan, K.F. (2005).** Arabic GramCheck: A grammar checker for Arabic. *Software: Practice and Experience*, 35(7), 643–665.
21. **Swedish Institute of Computer Science. (1988).** SICStus prolog user's manual.

Bassam H. Hammo received his B.Sc. in Computer Science from the University of Jordan, Amman, Jordan, in 1987. He has M.Sc. in Computer Science from Northeastern University and Ph.D. in Computer Science from DePaul University, Chicago, IL, USA (2002). He is an Associate Professor of NLP at the Department of Computer Information Systems and has been with the University of Jordan since 2003. His research interest is Arabic natural language processing and its applications. He leads the ANLP research group at the University of Jordan.

Asma Moubaiddin received her B.A. in Language and Literature from the University of Jordan, Amman, Jordan, in 1975. She received her M.A. in Applied Linguistics in 1986 and Ph.D. in Language and Linguistics from University of Essex, England, U.K., in 1992. She is an Assistant Professor at the Department of Linguistics and she has been with the University of Jordan since 2004. Her research interests include syntax, semantic, language acquisition and knowledge sharing and dialog systems.

Nadim Obeid received his B.Sc. in Mathematics in 1979 and B.Sc. in Business Administration in 1980 from Lebanese University, Lebanon. He has M.Sc. in Computer Studies and obtained Ph.D. in Computer Science from University of Essex, England, U.K., in 1987. He is a full time Professor at the Department of Computer Information Systems and has been with the University of Jordan since 2004. His research interests include knowledge representation, multi-agents and dialog systems.

Abeer Tuffaha received her B.Sc. in Computer Information Systems in 2006 from the University of Jordan, Amman, Jordan. She obtained M.Sc. in Computer Information Systems from the University of Jordan, Amman, Jordan, in 2010. She has been a Senior Software Engineer with Souq.com in Amman, Jordan, since 2013. Her research interests include Arabic syntax and semantics.

Article received on 06/01/2014, accepted on 06/02/2014.