An Efficient Recursive Transition Network Parser for Arabic Language

Bilal M. Bataineh, Emad A. Bataineh

Abstract - Parsing Arabic sentences is a difficult task; the difficulties come from several sources. One is that sentences are long and complex, the other difficulties come from the sentence structure. The syntactic structure of sentence parts may be missing, taking different orders of words and phrases. The present work aims to develop an Arabic Parser. A new parser has been developed with the aim of analyzing and extracting the attributes of Arabic words. The parser has been written using top-down algorithm parsing technique with recursive transition network, the parser development was a two-step process. In the first step, the set of rules used in the study for Arabic parser have been generated from an existing Arabic text taught in k-12 grade levels. The second step was the implementation of the parser which analyses an Arabic sentence and determines if the sentence follows a valid grammatical structure. The parser has been evaluated against real sentences and the outcomes were very satisfactory.

Keywords: Parser, Lexicon, Arabic Language, Recursive Transition Network grammar

I. Introduction

Natural Language Processing (NLP) has many definitions that all share in dealing with natural language by using computers, Drake, M. in 2003 defined Natural Language Processing as a theoretically motivated range of computational techniques for analyzing and representing naturally occurring texts at one or more levels of linguistic analysis for the purpose of achieving human-like language processing for a range of tasks or applications [1]. The Applications of NLP include a number of fields of studies, such as information retrieval (IR), machine translation (MT), Question Answering (QA), text to speech (TTS) text summarization (TS), and so on. Information retrieval is one of Natural Language Processing Applications that appears obviously during these definitions, Information retrieval is a zstorage, searching, and retrieval of information [2].

II. Recursive Transition Network Grammars

Transition Network Grammars is formalism for representing grammars based on the notion of a transition network consisting of nodes and labeled arcs. It developed out of the concept of the transition network of a finite-state automaton, but is equivalent to push-down automata because the arcs comprising the network of a Transition Network Grammar represent transcriptions of the rules of a context-free grammar [4]. Sentences generated by the grammar are accepted by a Transition Network Grammar through the process of traversing the network comprised of these arcs. RTNs were first used to parse the syntax of natural language phrases [6].

Bilal M. Bataineh is with the Arb Academy for Banking & Financial Sciences, Amman, Jordam; e-mail: Bilail.Bataineh@aabfs.edu.jo.
Emad A. Bataineh is with Zayed University, Dubai, UAE; e-mail: Ema.Bataineh@zu.ac.ae.

RTNs are considered as development for finite state automata with some essential conditions to take the recursive complexion for some definitions in consideration. The grammar model described in figure 1 is called RTN because the arcs of the grammar can call other levels of the network to recognize subordinate constituents that can in turn call other levels (recursively).

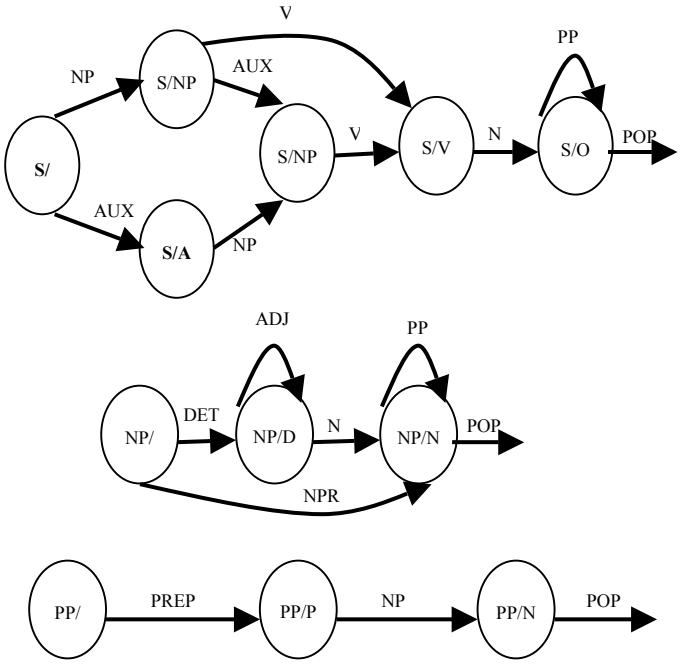

Figure 1: recursive transition network

III. Parsing Natural Language Processing

Parsing is about discovering a structure in an input, based on external information known about the elements of the input and their order. Generally, the external information consists of a lexicon, which is a list of input words, and a grammar, which describes which structures, may be built from, and implied by, sequences of words.[13] Parsing is the process of analyzing an input sequence in order to determine its grammatical structure with respect to a given formal grammar [3]. Natural language parsing aims to identify the syntactic structure of sentences. Full compositional parsing generates complete parse trees for input sentences (similar to parse trees. of context free languages), but also requires a training corpus annotated by

complete parse trees. The aim of the parsing process is to obtain an edge that covers all the input and that is labeled by the distinguished symbol.

A. Top-Down Parser

A top-down parser starts with the S symbol and attempts to rewrite it into a sequence of terminal symbols that matches the classes of the word in the input sentence. The state of the parse at any given time can be represented as a list of symbols that are the results of operations applied so far, called the symbol list. For example, the parser starts in the state (S) and after applying the rule **S → NP VP** the symbol list will be **(NP VP)**. If it then applies the rule **NP → ART N**, the symbol list will be **(ART N VP)**, and so on [5]. On a few words the parser starts from the largest elements (sentence) and breaks them down into incrementally smaller parts (words). Top-down parser search for a parse tree by trying to build from the root node S down to the leaves.

B. Bottom-Up Parsing

The basic operation in bottom up parsing is to take sequence of symbols and match it to right-hand side of the rules. You could build a bottom up parser by formulating the matching process as a search process. The state would simply consist of a symbol list, starting with the words in the sentence [5]. On a few words the parser starting with the smaller parts (words) and building towards some high level structure (sentence). In bottom-up parsing, the parser starts with the words of the input, and tries to build trees from the words up.

C. Top-Down Parsing With Recursive Transition Network

Li. W, et al. in 1990 presented a practical method for parsing long English sentences of some patterns. The rules for the patterns are treated separately from the augmented context free grammar, where each context free grammar rule is augmented by some syntactic functions and semantic functions. The rules for patterns and augmented context free grammar are complimentary to each other. This method bases on some patterns of long English sentences. The patterns can be inserted in the lexicon or the augmented context free grammar to guide the parser [14].

IV. Definition of Arabic Language

Arabic language is one of the most popular languages in the world, it is the official language of twenty two Middle East and African countries, and is spoken by more than 200 millions of people all over the world [7] . Arabic is the language of the Quran (the sacred book of Islam). As the language of the Qur'an, it is also widely used throughout the Muslim world [8]. It is belongs to Semitic group of language, unlike English language which belongs to the Indo-European language group [7]. There are many Arabic dialects, which are classified into three classes: Classical, Modern Standard Arabic and Local dialects [8].

Classical Arabic - the language of the Qur'an - was originally the dialect of Mecca in what is now known as Saudi Arabia [8].

Modern Standard Arabic- it is an adapted form Classical Arabic, which is used in books, newspapers, on television and radio, in the mosques, and in conversation between educated Arabs from different countries [8].

Local dialects- it is different from country to other, it is difficult to understand between the people of different countries, a Moroccan might have difficulty understanding an Iraqi, even though they speak the same language [8].

Arabic language has 28 letters, 25 of them consonants and three vowels " **ا, و , ي** ", which can be short or long. 12 of them are unique to Arabic language, which does not have any corresponding English letters language such as " **ح، خ، ض، ص**" consonant and difficult for foreigners to pronounce exactly [8]. In addition, the letters are divided into categories according to basic letter shapes, and the difference between them is the number of dots on, in or under the letter. Dots appear with 15 letters, of which 10 have one dot, 3 have two dots and 2 have three dots. In addition to the dots, there are diacritical marks that contribute phonology to the Arabic alphabet **[**9**].**

Arabic script is not like English script where Arabic can only be written in cursive script, each letter in Arabic has many different shapes depending on whether it is in the beginning or in the middle or at the end of the word. Arabic has diacritical marks called "harakat al- tashkeel" which can be placed above or under the letters. These diacritical marks are very important within the Arabic script not only to have the right meaning of that word. In addition to that, these diacritical marks could change the pronunciation of the letters from one to another.

The grammatical system of the Arabic language is based on a root-and-pattern structure and considered as a root-based language with more than 10,000 roots **[**10**]**. A root in Arabic is the bare verb form which can consist of three letter (trilateral), which is the majority of Arabic words, four letter (quadrilateral) , five letter (pent literal), or six letter (hex literal), each of which generates increased verb forms and noun forms by the addition of derivational affixes [11]. Affixes in Arabic are prefixes, suffixes and infixes. Prefixes are attached at beginning of the words, where suffixes are attached at the end of the word, and infixes are found in the middle of the words, for example, the Arabic word "**المدرسات** " which means "women teachers", consists of the following elements:

Word	Prefix	Suffix	Infix	Root
المدرسات	الم	ات	-	درس

A. Arabic Words Classification

Arabic grammarians traditionally classify words into three main categories: nouns, verbs, and particles. All verbs in Arabic and most of the nouns are derived from the root verbs. These categories are also divided into subcategories, which collectively cover the whole of the Arabic language. These categories are:

Noun : A noun in Arabic is a name or a word that describes a person, thing, or idea, the linguistic attributes of nouns are (Gender, Number, Person, Case, and Definiteness) [12].

Verbs Verbs indicate an action, although the more on action and aspects are different. Verb categories are divided into subcategories such as Perfect, Imperfect, and Imperative. The verbal attributes are (Gender, Number, Person...) [12]

Particle The Particle class includes: Prepositions, Adverbs, Conjunctions, Interrogative Particles, Exceptions and Interjections [12]

B. Arabic Sentence Structure

A Text in Arabic language is composed of set of sentences, these sentences might be a verbal sentences (الجملة الفعلية), which has the structure verb-subject-object, and must start or with a verb, it might be a nominal sentences (الجملة الاسمية), which must start with a noun. [7] In each case the sentence is either simple or compound. The difference between the simple sentence and the compound sentence is that the former does not have a complementary that could occur at the end of the sentence.

Verbal Sentences

The verbal sentence is the sentence that begins with the verb followed by the subject and then the predicate, for Example: **كَتَبَ الطَّالبُ الدَّرْسَ**

The verb (كَتَبَ) (/kataba/ = he wrote = past)
The subject (الطَّالبُ) (/at-talebu/ = The student)
The Accusative Object (الدَّرْسَ) (/addarsa/ = the lesson)

When the subject is unknown, we change the verb form, for example: كَتَبَ (/kataba/) (Active Verb) كُتِبَ (/kutiba/) (Passive verb). We say: (/kutiba-d-darsu/) (كُتِبَ الدَّرْسُ) "the lesson is written" in past tense, in the present tense : (/yuktabu-d-darso/) (يُكْتَبُ الدَّرْسُ)

There are two types of verbs:

The intransitive Verb : that needs only his subject , for example: (جاء الوَلَد) "the boy comes"
The transitive Verb: that needs his subject and an accusative object, (المَفْعول بِهِ), for example: (كَتَبَ التِّلْميذُ الدَّرْسَ)

For the three elements of verb, subject and object, there are different word orders for Subject and Object [8]:

Verb + Subject + Object**:**
Example: (**أَكَلَ الوَلَدُ التُّفاحَة**) **(**The boy ate the apple**)**.
Verb + Object + Subject:
Example: (**أَكَلَ التُّفاحَةَ الوَلَدُ**) **(**The boy ate the apple**).**

Nominal Sentences

A nominal sentence (**الجملة الاسمية**) always starts with a noun or a pronoun, It has two parts. The first part is the subject of the sentence and is called (**مبتدأ**), and it is what the speaker is speaking about, and the second part is the predicate and called (**خبـر**), which is give information about the (**مبتدأ**) subject [61]. The subject (**مبتدأ**) is usually placed at the beginning of a sentence. It can some times occur at the end instead, such a subject (**مبتدأ**) is known as a delayed subject (**مبتدأ مؤخر**) [61]. The subject (**مبتدأ**) may be a single noun or a pronoun or a phrase but it cannot be a verb, the subject (**مبتدأ**) is always in the nominative case i.e., the last letter takes a single (**ضمة**) (ُ) if definite - with definite article (**ال**) - and takes (**تنوين ضم**) (ٌ) if indefinite - without the definite article (**ال**) [8].

V. Structure of the Proposed Arabic Parser

Our main goal was to implement a computer system to parse Arabic sentences. The architecture of the system is given in Figure 2 In this architecture, the boxes are indicating the processes of the system and the arrows indicate the flow of information between system parts. As shown, the input of the parser is the words of the input sentence with their features. These features are retrieved from the lexicon.

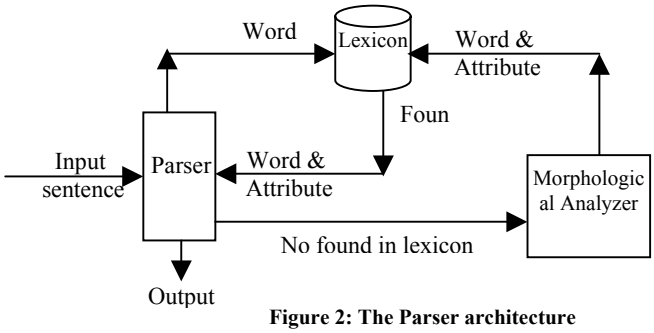

Figure 2: The Parser architecture

A. Arabic grammar

The syntax derivation process or forms of Arabic sentences is a very complex process without identifying a specific domain, in which we can derive this grammar. as well as the characteristics of Arabic language increase the difficulty of derive this grammar, especially when you face deceiving some portions of Arabic sentence (hidden pronouns), also in delaying and posting processes, between sentence parts. A big variation between the sources of Arabic syntax were found. An Arabic sentence structure differs according to the domain and the time age in which it is used, for example, sentence structure in poetry different from sentence structure in discourses, letters and proses.

In our approach, we analyzed the text and identified the different patterns of a sentence. With the help of Arabic linguistics, we derived the Arabic grammar rules from these patterns. Our objective is to formulate the grammar of simple Arabic sentences. Here, we try to derive grammars for sentences which mostly used in Arabic language to cover large items of Arabic grammar. This process has found two forms for Arabic language sentence; they are simple sentence which does not connect with another sentence in meaning. Another form is compound sentences which is

more than one simple sentence connected by conjunctions (أدوات عطف).

Arabic sentence has more than one type, the first type is nominative sentence that starts with name, the second one is verbal sentence that starts with verb, There are other kinds like sentences which begin with especial verbs (كان وأخواتها , كاد وأخواتها) or begin with especial articles like (إن وأخواتها) also, there are interrogative sentences which begin with question (أدوات استفهام) ,and other types of sentences. Through researching and briefing we find that their are hundreds of Arabic rules which depends on subjects which are used in Arabic language like literary works and scientific fields and others, then we have acquired the sentences from school materials which are familiar among educated people, we extracted the grammar rules from the sentences extensively used, some these rules are indicated in table 1.

Table (1): Rules of some nominal sentences

1.	**S**→NP	ج ← م ا
2.	**S**→NP+PP	ج ← م ا + ج م
3.	**S**→NP+PP+V+PP	ج ← م ا + ج م + ف + ج م
4.	**S**→NP+PP+V+NP	ج ← م ا + ج م + ف + م ا
5.	**S**→NP+V+PP	ج ← م ا + ف + ج م
6.	**S**→NP+AP	ج ← م ا + م ض

B. Implementation of the Parser

Arabic parser implemented using Top-Down parsing with recursive transition network algorithm, also agreement features are used to ensure the correction of syntax structure of the Arabic sentences, Agreement features are divided into two categories:

Gender agreement:

The first category in this agreement case is gender (male and female), it is very important attribute in Arabic language; it must be corresponded between some sentence words (verb, subject adjective…), the complete sentence is rejected because of the sentence do not agree with some features constraints. For example,

Sentence 1: **ذهب الطالبة** **"**the student went**".**

Sentence 2: **ذهب الطالب** **"**the student went**".**

The first sentence is incorrect; there are no agreement between the verb "**ذهب**" which is **male** and the subject "**الطالبة** " which is **female**, but the second sentence is correct, both verb and subject are **male**.

Number agreement:

The second category is number (singular, dual and plural), in this agreement the influence of number attribute is the same as the gender attribute, in which if there is no agreement between some parts of a sentence in number leads to be not acceptable by parser. In contrary the gender, the order of verb and subject in the sentence affects the number agreement; if the verb precedes the subject then the agreement is not necessary, but if the subject precedes the verb, the verb must agree with the subject in number.

VI. Parser Evaluation

A quazi experiment was conducted to assess the effectiveness and efficiency of the new parser. The purpose of the experiment was to test whether the parser is sufficiently for application to real Arabic sentences or not. Various an unrestricted Arabic sentences were selected from Grade-6 Arabic textbook.

A. Findings

In this section we discuss the testing results whether the input sentence is parsable. Table (2) shows the results of the parser. These results fall into two categories: the parsable sentence and the unparsable sentence.

The **parsable** sentence is divided into two subcategories:

1. **Syntactical Correct**: Which has led to a complete successful parse of the input sentence? For example, the input sentence (**تذهب الطالبة إلى المدرسة)** is syntactical correct sentence.
2. **Syntactical Incorrect**: Which has led to complete parsing of the input sentence but the result is a syntactical incorrect structure; the source of this error is not match in the attributes (gender, number) between words of sentence, For example, the input sentence (**يذهب الطالبة إلى المدرسة) i**s not parsed by our parser because the subject (الطالبة) takes the feature gender as female, but the prefix (ي) of the verb (يذهب) of the sentence indicates that this feature value is for male.

The **unparsable** sentence is divided into subcategories:

1. **Lexical problem**: in which the parser does not find the word in the lexicon.
2. **Incorrect sentence**: This has failed to parse because the input sentence is incorrect
3. **Failure**: the sentence which is not recognizable by linguists according to Arabic grammar rules

Table (2): results of the parser

		# of sentences	**percentage**
Parsable sentence	Syntactical Correct	77	85.6 %
	Syntactical Incorrect	2	2.2 %
Unparsable sentence	Lexical problem	4	4.4 %
	Incorrect sentence	2	2.2 %
	Failure	5	5.6 %
Total		90	100 %

The total number of sentences used in the test was 90. The sentence length was arranged 6 words. The result shows

that the number of sentences parsed successfully was 77 sentences, about 85.6%, 2 sentences were Syntactical Incorrect, about 2.2%. The number of sentences that were not parsed (has Lexical problem) was 4 sentences, about 4.4%.The number of sentences that were not parsed (Incorrect sentence) was 2 sentences, about 2.2%.The number of sentences that were not parsed (not recognizable by linguists according to Arabic grammar rules) was 5 sentences, about 5.6%

B. Analysis and Discussion of results

Analysis of Incorrect Syntactical Sentences

Recall that the number of syntactical incorrect sentences was 2 sentences. The parser assigns incorrect result to the input sentence. In other words, the parser complete sentence parsing but the result is incorrect, this result due to incomplete agreement between words attributes (gender, number).

Analysis of Unparsable sentences

Recall that the number of unparsable sentences was 11 sentences. The parser fails to assign any rule to input sentence. These are classified into three categories:

Lexical problem: The parser fails to assign any rule to input sentence, because some parts of sentences are not available in the lexicon, so the parser does not get the attributes of these parts.
Incorrect sentence: The parser fails to produce a rule for input sentence because the syntactic form of the sentence is not correct, on other words; it is impossible to find equivalent rule to the sentence form in the parser
Failure: The parser fails to produce a rule for input sentence because the syntactic form of the sentence is not included in the grammar. This means that failure may fulfill when the sentence structure is correct.

VII. Conclusion

The main objective of this study is to design, build and evaluate prototype system for parsing Arabic sentences and determine if these sentences syntactically correct or not. Arabic language lacks parsing systems for analyzing Arabic sentences. Parsing systems became very important in Natural language processing because it is used as a first step in the most of Natural language processing applications. Moreover, this system can be widely used for educational purposes. Parsing Arabic sentences is a difficult task. In Arabic natural language processing, there are no predefined forms for analyzing sentences, which makes parsing problematic. The Arabic sentence is complex and syntactically ambiguous due to the frequent usage of grammatical relations, conjunctions, and other constructions

The methodology was mainly based on studying and analyzing the grammar of Arabic language conforming to gender and number, formulize the rules using context free grammar, representing the rules using transition networks, constructing a lexicon of word that will be in sentences structure, implementing the recursive transition network parser and evaluating the system using real Arabic sentence.

A top-down algorithm parsing technique with recursive transition net-work was used in the parser development, The efficiency of the developed parser has been evaluated, A sample of 90 sentences was used in the test. The result shows that 85.6% of sentences were parsed successfully, 2.2% of sentences were parsed unsuccessfully and 14.4% of sentences not parsed for various reasons, 4.4% Lexical problem, 2.2% Incorrect sentences, 5.6% not recognizable by linguists according to Arabic grammar rules In conclusion, the parser was an efficient and produces satisfactory results.

REFFERENCES

[1] Drake. M, (2003), Encyclopedia of Library and Information Science, CRC Press.

[2] Aslam, J. et al. (2003). Challenges in Information Retrieval and Language Modeling. ACM SIGIR Forum, 37(1):31-47.

[3] Istek, O. (2006): A Link Grammar For Turkish, M.S. Thesis, Institute Of Engineering And Sciences Of Bilkent University

[4] Woods. W, (1970) : Transition Network Grammars of Natural Language Analysis, Communications of the ACM, 13, 591-606. Reprinted in Barbara J. Grosz, Karen Spark Jones, and Bonnie Lynn Webber (eds.) Readings in Natural Language Processing. Los Altos, USA: Morgan Kaufmann, 1986, pp. 71-87.

[5] James Allen, J, (1995) : Natural Language Understanding. Benjamin/ Cummings Publishing Company, Inc.

[6] Stehno, B And Retti, G. (2003): Modeling the logical structure of books and journals using Augmented Transition Network Grammars. In: Journal of Documentation, Vol. 59 No. 1 p. 69-83.

[7] Al-Shalabi, R., Kanaan, G., & Al-Serhan, H. (2003): New Approach For Extracting Arabic Roots. Paper presented at the International Arab Conference on Information Technology (ACIT'2003), Alexandria, Egypt. pages 42-59, Alexandria, Egypt.

[8] Al-Shalabi, R., Kanaan, G., & Al-Qraini, S. (2004): An Automatic System For Extracting Nouns From A vowelized Arabic Text, In ACIT 2003,Egypt

[9] Abu-Rabia, S. & Awwad, J, (2004): Morphological structures in visual word recognition: The case of Arabic. Journal of Research in Reading, Volume 27, Issue 3, pp 321–336.

[10] Ali, N. (1988): Computers and Arabic language. Egypt: Al-Khat Publishing Press, Ta'reep. 64

[11] Saliba, B., & Al-Dannan, A. (1990): Automatic Morphological Analysis of Arabic: a study of Content Word Analysis. Proceeding of the First Kuwait Computer Conference: 231-243.

[12] Kanaan G., Al-Shalabi R., Sawalha M., (2003): Full Automatic Arabic Text Tagging System, the proceedings of the International Conference on Information Technology and Natural Sciences, Amman/Jordan, pp 258-267.

[13] Placeway, P. (2002). High-Performance Multi-Pass Unification Parsing. PhD thesis, Carnegie Mellon University, Pittsburgh, PA. Technical Report CMU-LTI-02-172.

[14] Li. W, et al., (1990): Parsing Long English Sentences with Pattern Rules: Proc. 13th International Conference on Computational Linguistics, pp.410–412, Helsinki, Finland.