

A semi-automatically generated TAG for Arabic: Dealing with linguistic phenomena

Cherifa Ben Khelil^{1,2}, Chiraz Ben Othmane Zribi¹, Denys Duchier², and
Yannick Parmentier³

¹ RIADI - ENSI Université La Manouba Tunisia

² LIFO - Université d'Orléans France

³ LORIA - Projet SYNALP Université de Lorraine France

cherifa.bk@gmail.com

chiraz.zribi@ensi-uma.tn

denys.duchier@univ-orleans.fr

yannick.parmentier@loria.fr

Abstract. Arabic is a challenging language when it comes to grammar production and parsing. It combines complex linguistic phenomena with a rich morphology that make its processing particularly ambiguous. This led us to choose the Tree-Adjoining Grammar (TAG) formalism. Indeed, TAG provides sufficient constraints for handling diverse linguistic phenomena and seems to be adequate to represent Arabic syntactic structures. In this paper, we present a semi-automatically generated TAG for modern standard Arabic using a compiler and a metagrammatical description language called XMG (eXtensible MetaGrammar). We focus on the linguistic coverage of our grammar, and show how we used TAG and XMG's properties to define in an expressive and concise way different linguistic phenomena. To check the coverage of our grammar, we have set up a development environment including a parser and using a test corpus of linguistic phenomena gathering both grammatical and ungrammatical sentences.

Keywords: Tree adjoining grammar (TAG); metagrammar; parsing; corpus of linguistic phenomena; Arabic language.

1 Introduction

Arabic is a challenging language when it comes to grammar production and parsing. It exhibits specific features such as free word order, combined with a rich morphology and the omission of diacritics in most of written texts. These linguistic phenomena affect the syntactic parsing process and make it more difficult. The parsing task requires an amount of knowledge and resources that provide information about the correct structural representations of the input data (text or sentence). Indeed, the process of analysing must be conforming to the rules of a formal grammar. Most of methods for parsing adopted the rule-based approach, which uses well-defined formal grammars to represent the Arabic syntax. Among them [1], [2] and [3] offer a syntactic analysis based on Head-Driven

Phrase Structure Grammars (HPSG) formalism. [4] developed a grammar in Lexical Functional Grammar (LFG) to parse Arabic. [5] and [6] used Context Free Grammar (CFG) and [7] a Unification Based Grammar (UBG). However, grammars creation takes a lot of time [4] since they were encoded manually. In addition, it is difficult to have a grammar that covers all the syntactical structures of a language. To date there is not a wide-coverage grammar of the Arabic language. In this context, many efforts have been put into semi-automatic grammar production, either by acquiring grammar rules from annotated corpora [8] or by using description languages to capture generalizations among these rules. The latter permits to formally specify the structures of a target grammar and it is considered as a grammar specification (called metagrammar) that can be compiled into an electronic grammar. This grammar production technique has been used to develop several electronic grammars for French [9], English [10] and German [11]. However, it was not applied for Arabic. This work is the first attempt to create a grammar for Arabic by the means of a meta-grammar. We adapted the XMG description language [12] for Arabic to semi-automatically generate a Tree-Adjoining Grammar (TAG) [13] called ArabTAG V2.0. Our choice of TAG was motivated by its power of representation (i.e. simple, complex, combinatorial, shared structures) and its ability to deal with certain phenomena that are specific for Arabic.

The paper is organized as follows. In section 2, we present the grammatical resources used in the generation process of our grammar. Section 3 shows how we used TAG and XMG's properties to deal with specific linguistic phenomena of Arabic. Section 4 describes a corpus of phenomena that we built to check grammar coverage. Finally, section 5 gives an overview of our grammar's coverage.

2 Semi-automatically generating ArabTAG V2.0

Tree Adjoining Grammar (TAG)[14] is a syntactic formalism that handles the links between the constituents of the sentence to build grammatical representations. It consists of a set of elementary trees divided in initial tree (also called substitution nodes and are marked with the symbol \downarrow) and auxiliary tree (has a "foot node" marked with the symbol $*$). The two compositions operations authorized by TAG are substitution and adjunction. The resulting tree obtained by the end of these operations is called a derived tree. Substitution appends a frontier node with another tree whose top node has the same symbol. Adjunction is more powerful since it allows inserting an auxiliary tree into the center of another tree. We cannot, assert that this formalism is undoubtedly the best to represent Arabic. Nevertheless, its characteristics make it possible to represent specific syntactic structures and frequent phenomena in Arabic. To our knowledge, there are very few TAG-based descriptions of Arabic. The first TAG by Habash and Rambow [8], was extracted from an Arabic Treebank (namely the Penn Arabic TreeBank – PATB). The corpus they used is the Part 1 v 2.0 of PATB [15] [16]. The second is a handcrafted tree-adjoining grammar named

ArabTAG (Arabic Tree Adjoining Grammar)[17]. Our work takes its origins from the latter. This grammar describes different syntactic components of different levels: sentences, phrases and words, as well as the various information related to them (morphological and syntactic information). ArabTAG has feature based structure and is semi-lexicalized. It contains two sets of elementary trees: 35 lexicalized trees (reserved to prepositions, modifiers, conjunctions, demonstrative pronouns) and 215 patterns trees (represent verbs, nouns, adjectives or any kind of phrases). The construction of these structures was based on school grammar books and books of Arabic grammar [18]. The current version of this grammar has some limitations that can be summarized as follows:

- Minimal coverage of syntactic structures. Structures enriched with supplements (i.e circumstantial complements of time, place) are not described.
- The representation of forms of agglutination is not well reflected.
- The grammar emphasizes syntactic relations without regard to semantic information.
- ArabTAG is not organized in a hierarchical way, which does not facilitate grammar extension and maintenance.

We have proposed a new version ArabTAG V2.0 [13] that considers the aspects mentioned above. We used XMG (eXtensible MetaGrammar) description language [12] to describe Arabic for it exhibits particularly pertinent features:

- it is highly expressive, since it defines highly factorized grammar descriptions.
- it is particularly adapted to the description of tree grammars and has been used to develop several electronic TAG grammars for e.g. French [9], English [10], German [11].
- it is highly extensible and can be configured to describe various levels of language, such as semantic or morphology.

We have semi-automatically generated ArabTAG V 2.0 (see Figure 8) from a reduced description of grammar rules. First, the metagrammatical language XMG is used to define Arabic-XMG meta-grammar. It is described as (conjunctive and disjunctive) combinations of tree fragments. Such fragments are defined as formulas of a tree description logic based on dominance and precedence relations between node variables. We refer the reader to [13] for additional information about Arabic-XMG meta-grammar. Then, this compact description is automatically compiled into the ArabTAG V2.0 grammar by the XMG2⁴ compiler [19]. Afterward, we extended our meta-grammar by associating semantic information with the defined families of elementary trees to make the interfacing easier between syntax and semantic (not described here, please see [20]).

3 Dealing with syntactic phenomena in ArabTAG V2.0

In this section, we focus on presenting specific linguistic phenomena that are handled by ArabTAG V2.0.

⁴ XMG2 extends XMG by including a meta metagrammar compiler

3.1 Free word order

Arabic has a relatively free word order. Usually nominal sentences begin with a noun or a pronoun, while verbal sentences begin with a verb. The most used order in standard Arabic for a verbal sentence is VSO (V-verb, S-Subject, O-Object). It is possible to change the order of these components without altering the meaning of the sentence. TAG allows combining tree structures without taking into consideration the order of the combinations. Adjunction and / or substitution operations can be called in a free order and can produce sentences with multiple syntactic structures. Moreover, by using XMG's properties, we managed to deal with the semi-free word order within our metagrammar. To do this, we avoided imposing precedence constraints between nodes whose order change does not affect the consistency of the sentence. Let us consider the following sentence:

قرأ التلميذ الكتاب (reads the student the book). We can change the order of words to have the two combinations التلميذ قرأ الكتاب (the student reads the book) (SVO) and قرأ الكتاب التلميذ (reads the book the student) (VOS). As shown in Figure 1⁵, our grammar provides all tree models for these three combinations.

3.2 The adjunction of adverbs and optional complements

Adjunction in TAG allows the insertion of a complete structure at an interior node of another complete structure. It appears to be a natural way of handling adverbs and optional complements in natural language. In Arabic, adverbial object, circumstantial complement of time, circumstantial complement of place, causative object, etc. can be freely interspersed between arguments. We needed to provide two appropriate adjunction points for them: AG (adv_g) and AD (adv_d). AG is an adjunction point allowing an adverb (or optional complements) at the front of the clause and AD allows inserting an adverb (or optional complements) after a verb or an argument. For example, we can add the adverb كثيراً (a lot) in the sentence ينام علي (sleeps Ali) before the subject ينام كثيراً (sleeps a lot Ali) or after the subject ينام علي كثيراً (sleeps Ali a lot) as shown in Figure 2.

3.3 The representation of agglutination forms

The phenomenon of agglutination consists of joining proclitics and / or enclitics to simple forms of words, which gives rise to more complex forms called agglutinated forms. Proclitic is attached to the beginning of another word (coordinating conjunctions, prepositions, preverbal clitic, etc.). As for enclitic, it is at the end of the word (pronouns, anaphora, etc.). We can also have a sentence consisting of one agglutinated word as in the following example: سيكتبه (he will

⁵ In order to decrease the size of the figures some features have been omitted

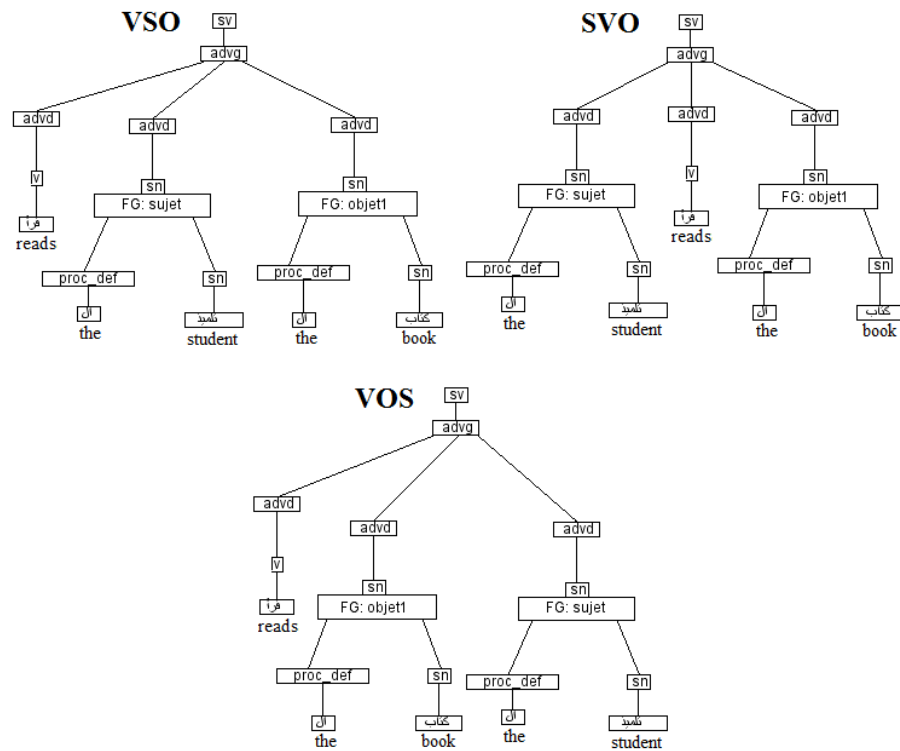


Fig. 1. Free word order of the sentence **قرأ التلميذ الكتاب** (reads the student the book)

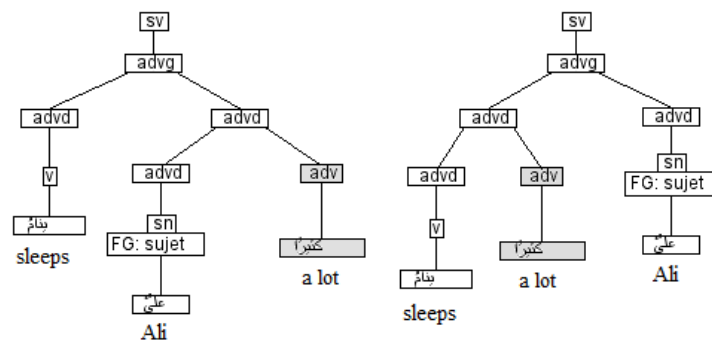


Fig. 2. Adding the adverb **كثيرا** (a lot) in the sentence **ينام علي** (sleeps Ali)

write it) which is composed of a particle of the future س (will), a verbe يكتب (he writes), and an object ه (it) that are all included in the same text form. To parse an agglutinated form, we must proceed to its division into proclitic / radical / enclitic. This division is itself confronted with a problem of ambiguity since for a single lexical unit we can have several possible divisions. TAG makes it possible to treat this phenomenon thanks to a finite set of possible feature structures [21] associated with the nodes of its elementary trees. These structures contain morphological and syntactic information, which help to assist the parsing procedure and thus remove the ambiguities that may arise. For the example above (see Figure 3), we can define in the same feature structure that the proclitic س (will) is a particle of the verb (feature pos: proc.v). This kind of particles can only be attached to the verb in the indicative mode (feature mode: ind). We can notice that the mode of the verb كتب (to write) to which this particle is attached is indeed in the indicative mode يكتب (he writes). Lastly, the enclitic attached to the end of the verb represents its object (feature fg: objet1) with accusative case (feature cas: acc).

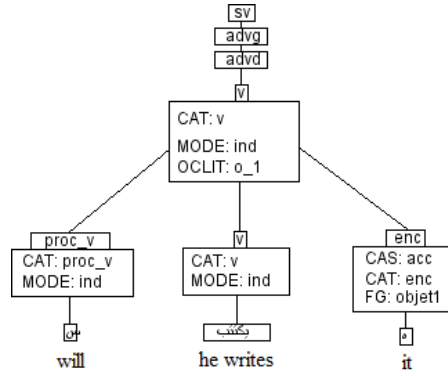


Fig. 3. Derived tree of the sentence سيكتبه (he will write it)

3.4 Agreement rules

In Arabic, there are many agreement rules: between adjectives and nouns (in definiteness, gender, number, and case), between subjects and verbs and between pronoun and verbs. These rules are handled in our grammar by interesting morphosyntactic features involved in agreement at the appropriate nodes. These features are number, gender, person, case, definiteness and the feature of humanness (human or not). For example, in adjectival phrase, adjective agree with noun in definiteness (def), gender (gen), number (num), and case (cas). These

constraints are ensured with defined feature structures as shown in Figure 4. As illustrated, to get a correct adjectival phrase these features should be equal.

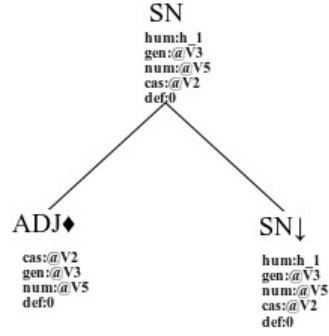


Fig. 4. Example of an elementary tree for adjectival phrase.

3.5 Embedded structures

Embedded structures, commonly known as relative and subordinate clauses, are very common in Arabic. Embedding is the process by which one clause is included in another. In this case, the length of a sentence is not limited and its segmentation is difficult. The representation of this phenomenon is possible with TAG thanks to the adjunction operation. Since adjunction allows inserting a complete structure in another structure, it makes embedding representation very natural. Moreover, it highlights recursion by allowing adding several embedded structures in the same sentence. Let us consider the following sentence

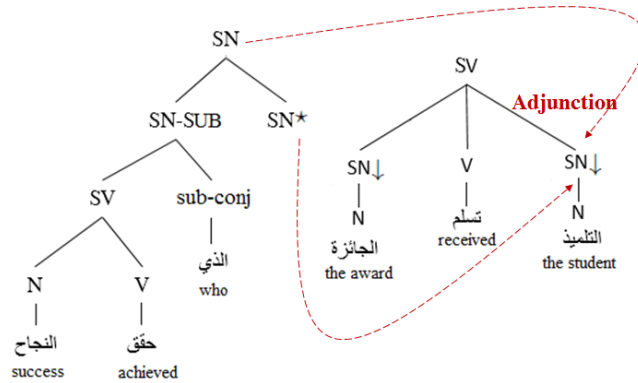


Fig. 5. Example of handling embedded structures with TAG.

التلميذ تسلمَ الجائزة (the student received the award) shown in Figure 5. We can add the subordinate clause الذي حقق النجاح (who achieved success) between the subject التلميذ (the student) and his verb تسلمَ (the award). The resulting sentence التلميذ الذي حقق النجاح تسلمَ الجائزة (the student who achieved success received the award).

3.6 Crossed dependencies

With adjunction, it is also possible to represent structures of complex sentences such as sentences containing crossed dependencies. This phenomenon occurs when dependency relations between two series of words cross over each other. Figure 6 shows an example of handling crossed dependencies using two adjunction operations.

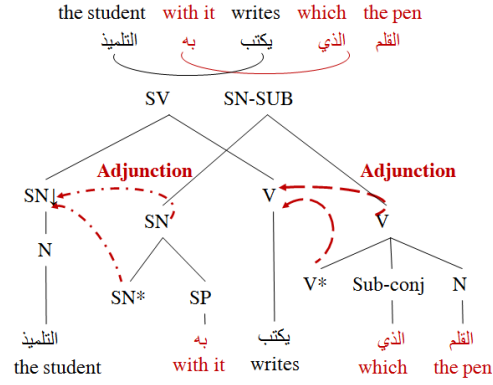


Fig. 6. Example of handling crossed dependencies with TAG

3.7 Subject Omission

In Arabic, subject may be implied or replaced by a pronoun (this makes it an elliptical clause). Our grammar covers this type of structures and offers the corresponding models to represent them. Figure 7 shows two sentences with the same meaning (He sleeps): (1) ينامُ composed of a verb and an elliptical subject and (2) هو ينامُ composed of a pronoun and a verb.

4 Building a corpus of phenomenon

In order to verify grammar coverage, we set up a development environment while designing ArabTAG with XMG (see Figure 8). We defined manually syntactic

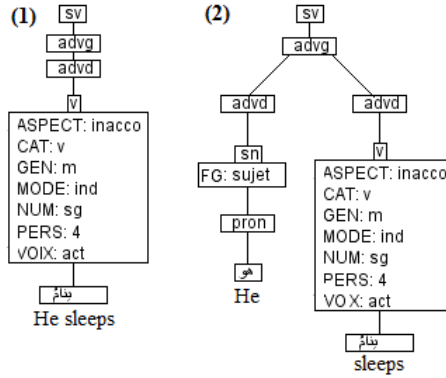


Fig. 7. Derived trees for *يَنَامُ* and *هو يَنَامُ* (he sleeps)

and morphological lexicons for Arabic following the 3-layer lexicon architecture of the XTAG project [22] as a proof of concept:

- A basis of tree schemas classified into families of elementary trees
- A lemma basis where each lemma is associated with one (or more) family trees
- A morphological basis in which each flexed form is associated with a lemma and its appropriate morphosyntactic information

The purpose of this validation is to evaluate and to reduce both under and over-generation. Our grammar must be able to recognize valid sentences that cover linguistic phenomena of Arabic (sentences described in schoolbooks, Arabic news, etc.) and to reject ungrammatical sentences. Each new syntactic phenom-

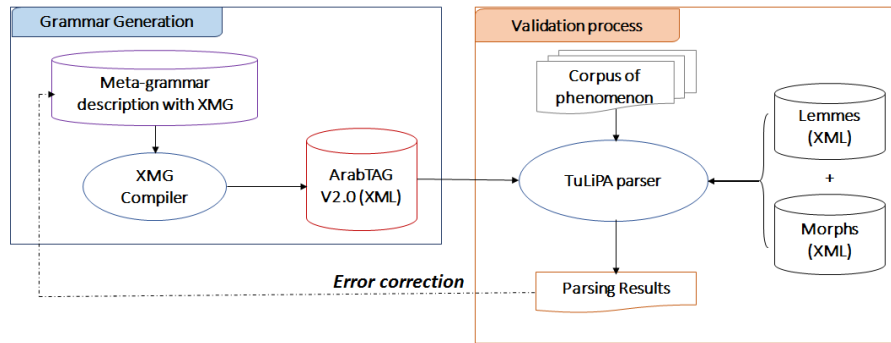


Fig. 8. Validation architecture of ArabTAG V2.0.[20]

ena included in ArabTAG V2.0 leads to the extension of a test corpus gathering

both grammatical and ungrammatical sentences. This corpus is called corpus of phenomenon. We used the TuLiPA parser [23] on the corpus to check the quality of the grammar. The parsing results help us to fix potential errors and bugs in our metagrammatical description and allow us to check the consistency of the defined TAG structures when it is extended. By the end of this verification, the corpus of phenomenon had 212 examples of phrases and sentences (150 grammatical sentences and 62 ungrammatical sentences). It contains 134 verbal sentences, 45 nominal sentences, 32 noun phrases and 1 prepositional phrase. Ungrammatical clauses were mainly added to check if the grammar could return syntactic configurations with incorrect agreement. The following table summarizes the different phenomena covered by our grammar:

Table 1. Phenomena covered by the corpus

Phenomenon	Number of sentences/phrases
Active forms	123
Adverbial object	6
Agglutination forms	26
Agreement rules	25
Circumstantial complement	9
Ditransitive verbs	67
Elliptical subject	17
Embedded structures	11
Free word order	44
Interrogative Sentences	10
Intransitive verbs	29
Passive forms	11
Transitive verbs	38

5 ArabTAG V2.0 coverage

So far, we have generated 668 trees from a description made of 29 classes (that is, 29 tree fragments or combination rules) as shown in Figure 9. The current version of the grammar covers verbal phrases (active and passive form), nominal sentences and phrasal structures. These latter have several types: noun phrase (مركب اسمي)⁶, subordinate phrase (مركب موصولي)⁷ and prepositional phrase

⁶ Has several categories: the annexation phrase (مركب إضافي); the adjectival phrase (مركب نعتي); the corroborative phrase (مركب توكيدي); the approbative phrase (مركب بدلي); the state phrase (مركب بحال المفردة); the conjunctive phrase (مركب شبه إسنادي) and the semi-propositional phrase (مركب عطف).

⁷ It begins with a subordinate conjunction or a relative pronoun and will be followed by a verb.

(مركب حرفي)⁸. In addition, ArabTAG V2.0 covers elliptical and subordinate structures. It takes into consideration the change of the order of the sentence’s components and the agglutinative forms. Furthermore, it contains elementary trees for the representation of additional complements such as circumstantial complement of time, circumstantial complement of place and adverbs.

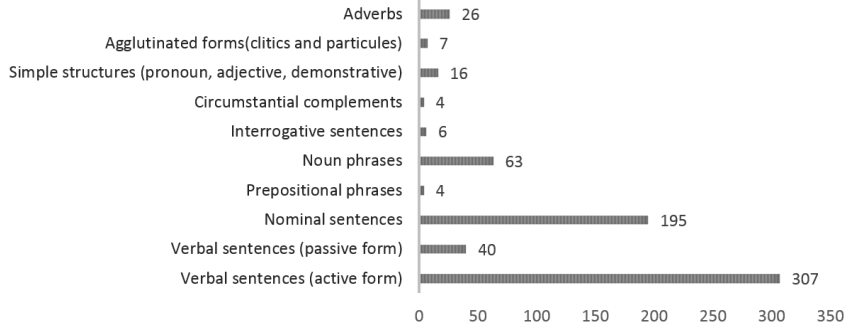


Fig. 9. Current tree distribution in ArabTAG V2.0

6 Conclusion

In this paper, we presented a Tree adjoining grammar for Arabic called ArabTAG V2.0. This grammar was produced semi-automatically using the meta-grammatical description language XMG. This method offers a relatively good control on the grammar being produced and allows its extension with various levels of description. Indeed, our produced grammar describes the syntax and the semantic levels. In this article, we focused on presenting only specific syntactic phenomena that are handled by ArabTAG V2.0. These phenomena make the syntactic parsing more difficult especially those considered complicated, such as embedded structures and crossed dependencies. We have shown that TAG formalism is suited for handling such phenomena. We have also built a corpus of phenomenon in order to verify grammar coverage while designing ArabTAG V2.0. The overall size of our grammar amounts to 668 trees, which correspond to the basic syntactic structures of Arabic sentences (verbal and nominal sentences) as well as the different phrasal structures (prepositional phrases and noun phrases). Our main perspective, in the near future, is to evaluate our grammar using a significant syntactic-semantic test corpus. Due to the unavailability of the resources necessary for such task, we started to build a test corpus larger than the corpus of phenomenon. Another possible perspective following this evaluation would be to further extend ArabTAG V2.0 improving its coverage.

⁸ It consists of a preposition followed by a noun (or a noun phrase).

References

1. Belguith, L. Aloulou, C. and Ben Hamadou.: MASPAP: De la segmentation à l'analyse syntaxique de textes arabes. CÉPADUÈS-Editions, editeur, Revue Information Interaction Intelligence I, Vol. 3, 9–36 (2007)
2. Loukam, M. and Laskri, M.T.: PHARAS: Une plateforme d'analyse basée sur le formalisme HPSG pour l'arabe standard: Développements récents et perspectives. JED'08, Journées de l'Ecole Doctorale, University Badji Mokhtar, Annaba, Algeria (2008)
3. Haddar, K. and Zalila, I. :An HPSG parser generation with the LKB for Arabic relatives. 3rd International Conference on Arabic Language. In proceedings of 3rd International Conference on Arabic Language Processing (CITALA'09). Rabat, Morocco (2009)
4. Attia, M.: Handling Arabic Morphological and Syntactic Ambiguity within the LFG Framework with a View to Machine Translation. Ph.D. Dissertation. University of Manchester, Faculty of Humanities (2008)
5. Bataineh, B. Bataineh, E.: An Efficient Recursive Transition Network Parser for Arabic Language. In proceedings of the World Congress on Engineering 2009 Vol II WCE 2009, July 1 - 3, London, U.K (2009)
6. Al-Taani, A. Mohammed, M. and Wedian, S.: A Top-Down Chart Parser for Analyzing Arabic Sentences. The International Arab Journal of Information Technology, Vol. 9, No. 3, March (2012)
7. Othman, E. Shaalan, K. and Rafea, A.: A Chart Parser for Analyzing Modern Standard Arabic Sentence. The MT Summit IX Workshop on Machine Translation for Semitic Languages: Issues and Approaches. New Orleans, Louisiana, U.S.A (2003)
8. Habash, N. and Rambow, O.: Extracting a tree adjoining grammar from the penn arabic treebank. Proceedings of Traitement Automatique du Langage Naturel (TALN-04), 277—284 (2004)
9. Crabbé, B.: Représentation informatique de grammaires fortement lexicalisées : Application à la grammaire d'arbres adjoints. Thèse de Doctorat, Université Nancy 2 (2005)
10. Alahverdzhieva, K.: XTAG using XMG. A core Tree-Adjoining Grammar for English. University of Nancy 2 / University of Saarland Master's Thesis (2008)
11. Kallmeyer, L. Lichte, T. Maier, W. Parmentier, Y. Dellert, J.: Developing a TTMCTAG for German with an RCG-based Parser. Published in The sixth international conference on Language Resources and Evaluation (LREC 08), Marrakech: Morocco (2008)
12. Crabbé, B. Duchier, D. Gardent, C. Le Roux, J. and Parmentier, Y.: XMG : eXtensible MetaGrammar. Computational Linguistics, 39(3):591—629 (2013)
13. Ben Khelil, C. Duchier, D. Parmentier, P. Zribi, C. and Ben Fraj, F.: ArabTAG : from a Handcrafted to a Semi-automatically Generated TAG, In TAG+12 : 12th International Workshop on Tree-Adjoining Grammars and Related Formalisms, Düsseldorf, Germany (2016)
14. Joshi, A. Levy, L. and Takahashi, M.: Tree adjunct grammars. Journal of Computer and System Sciences, 10(1), 136 — 163 (1975)
15. Maamouri, M. and Bies, A. Jin, H. and Buckwalter, T.: Arabic treebank: Part 1 v 2.0. LDC Catalog No.: LDC2003T06, ISBN: 1-58563-261-9, ISLRN: 333-321-196-670-5 (2003)

16. Maamouri,M. and Bies,A.: Developing an arabic treebank: Methods, guidelines, procedures, and tools. In Ali Farghaly and Karine Megerdooomian, editors, COLING 2004 Computational Approaches to Arabic Script-based Languages, pages 2–9, Geneva, Switzerland (2004)
17. Ben Fraj,F.: Construction d’une grammaire d’arbres adjoints pour la langue arabe. In Actes de la 18e conférence sur le Traitement Automatique des Langues Naturelles, Montpellier, France, June. Association pour le Traitement Automatique des Langues (2011)
18. Kouloughli,D. : La grammaire Arabe pour tous. Press Pocket (1992)
19. Simon Petitjean,S.: Génération Modulaire de Grammaires Formelles. Ph.D. thesis, Université d’Orléans, France (2014)
20. Ben Khelil,C. Ben Othmane Zribi,C. Duchier,D and Parmentier,Y. :A new syntactic-semantic interface for ArabTAG an Arabic Tree Adjoining grammar. In Proceedings of International Arabic Conference of Information Technology (ACIT 2017). Hammamet, Tunisia (2017)
21. Vijay-Shanker,K. Joshi,A.K.: Feature Structures Based Tree Adjoining Grammars. The 12th International Conference on Computational Linguistics (COLING ’88). Budapest, 22-27 August (1988)
22. XTAG Research Group,,:A lexicalized tree adjoining grammar for english, Technical Report IRCS-01-03, IRCS, University of Pennsylvania (2001)
23. Parmentier,Y. Kallmeyer,L. Lichte,T. Maier,W. and Dellert,J.: TuLiPA : A Syntax-Semantics Parsing Environment for Mildly Context-Sensitive Formalisms. In 9th International Workshop on Tree-Adjoining Grammar and Related Formalisms (TAG+9),121–128, Tübingen, Germany (2008)