RESEARCH ARTICLE OPEN ACCESS

Resources For Developing an Arabic Controlled Language

Hoyam Salah Elfahal Elebaed [1], Mohammed Kasbi [2], Mohammed Nasri [3], Karim Bouzoubaa [4], Khalid Tnaji [5]

[1] College of Post-graduate Studies, Sudan University of Science and Technology, Khartoum - Sudan
[2] Institute for Studies and Research on Arabization, Mohammed V University in Rabat - Morocco
[3] Laboratory of Science and Technology for the Engineer, National School for Applied Sciences, Khouribga, Sultan Moulay Slimane University - Morocco
[4] Computer Science Department, Mohammadia School of Engineers, Mohammed V University in Rabat - Morocco

ABSTRACT
Controlled Natural Languages (CNL) are a subset of natural languages with specific rules and vocabulary that aim to make communication clearer and more precise. Existing CNLs have been used for different purposes such as knowledge representation, interface language to the semantic web, translated and original English documents, etc. So far, many CNLs have been developed for Western languages, especially English, but no concrete CNL has yet been proposed for Arabic. In this paper, we introduce resources such as corpus and rules that help developing an Arabic CNL. We built an ACL corpus for Arabic from textbooks and websites dedicated to teach kids. We could extract in this first version of ACL 52 grammar rules and 551 sentences. We then encoded our ACL corpus using TEI format. Finally, and as a first use of our ACL, we developed and ACL syntactic checker that checks sentences according to ACL rules. Experiments have been conducted on three sets of sentences and showed an accuracy of 99.6% which is, for a first version, an excellent result.
Keywords: - Controlled Natural Language, Arabic CNL, ACL, ACL checker, Arabic CNL Checker, Arabic Corpus, and TEI.

I. INTRODUCTION

Controlled natural languages (CNLs) are a subset of natural languages that have specific rules and a specific vocabulary. In addition, they aim to be easily understood by both humans and computers. CNLs have been used for knowledge representation [1], interface language to the semantic web [2], translated and original English documents [3], and so on. Many CNLs have been developed for Western languages such as English [4], French [5] and German [6]. However, only one Controlled Language has been proposed for Arabic and aims to Authoring Ontologies [7] and three is no other proposed CNL for the Arabic language [8].

Traditionally, CNLs fall into two major categories according to the problem they are supposed to solve: 1) human-oriented CNLs such as Special English [9], Plain Language [10], CAA Phraseology [11], SMART Plain English [1] , Standard Language [12] and Easy English [13], and 2) machine-oriented CNLs such as Attempto Controlled English [14], PENG or Processable English [15], Common Logic Controlled English[2], Gellish [16][17], CLG [6] and INAUT [5]. According to Huijsen [18] and Kuhn [4], the first category is not necessarily based on a formal semantic and helps improve communication among humans, especially speakers with different native languages. Whereas the second category aims to improve (manual, computer-aided, semi-automatic, or automatic) translation or provide a natural and intuitive representation for formal notations.

In this paper, we introduce the first CNL for Arabic baptized Arabic Controlled Language or simply ACL. Obviously, developing an ACL will help in the design of advanced natural language services for Arabic users such as machine translation, more natural interfaces to the semantic web or more accurate named entity recognition systems [19]. ACL is designed as a machine-oriented CNL suited for general purposes and based on formal semantics.

To achieve this goal, we use a two-steps approach consisting in firstly building necessary resources (rules and corpus) and then in developing the ACL checker that syntactically checks and validates the correctness of texts according to this new ACL. The remainder of this paper is organized as follows. Section 2 shows how the proposed ACL is built. Section 3 presents the ACL checker software. Section 4 presents the conducted experiments as well as the corresponding results and section 5 concludes this paper.

[1] http://www.smartny.com/plainenglish.htm
[2] http://www.jfsowa.com/clce/specs.htm

II. RESOURCES FOR DEVELOPING ACL

Let us recall that a CNL is a subset of a natural language that needs to be limited in grammar rules and lexicon. The first step in developing the ACL should then be to define this subset and to construct a corpus of ACL sentences with their corresponding grammar.

A. *Building* the Arabic controlled language corpus

To build the ACL corpus, four textbooks and websites dedicated to teach Arabic language to kids are used: a) First grade book, Republic of Sudan[3] (كتاب الصف الاول جمهورية السودان), b) Al Jazeera Educational Site[4] (موقع الجزيرة التعليمي), c) Bella Preparatory School Girls Forum[5] (منتدى مدرسة بيلا الاعدادية بنات), and d) Albahr website[6] (موقع انا البحر). Table 1 shows the sources as well as the number of sentences extracted from each source.

TABLE 1: SOURCES USED FOR ACL CORPUS

Source	Original name	Number of sentences extracted from each source
First grade book, Republic of Sudan	كتاب الصف الاول جمهورية السودان	27
Al Jazeera Educational Site	موقع الجزيرة التعليمي	79
Bella Preparatory School Girls Forum	منتدى مدرسة بيلا الاعدادية بنات	33
Albahr	موقع انا البحر	25
Total		164

We use these sources since they begin with low level and easy grammar rules and vocabulary. From the content of these sources, we extract 164 sentences and get 76 Arabic grammar rules which are, for an ACL, a small number. To overcome this weakness, we extend these sentences with 700 new ones to get a total number of 864. For each sentence, some filters have been applied before integrating it in to our corpus. We firstly check if the suggested sentence is not already in the corpus with minor changes. For example, if the sentence " صَادَ غُلام غَزَالا." ("A boy caught a deer") already exists in the corpus, then another sentence with the object "sheep" instead of "deer" would be considered as already integrated in the corpus and would not be integrated again. Secondly, we set ourselves the goal that every sentence in our corpus must be correctly parsed. We then analyze all the sentences using the two known Arabic syntactic parsers Stanford [20] and FARASA [21]. After that, a linguistic verification was engaged on the output of these parsers. Table 2 shows the results of this step:

TABLE 2: ACL CORPUS CHECKING BY STANFORD AND FARASA PARSERS

Stanford Parser				Farasa Parser			
Parsed sentences	**Parsing failed**	**Incorrect Parsing**	**Correct Parsing**	**Parsed sentences**	**Parsing failed**	**Incorrect Parsing**	**Correct Parsing**
864	285	558	21 (2.4 %)	864	207	62	595 (68.9%)

For clarity, the parsing is considered failed when the parsing tool reports a technical error as result. When, in contrast, the parsing succeeds but the result is not correct from a linguistic point of view, this parsing is then considered as incorrect.

We can notice from table 2 that among 864, Stanford Parser fails in parsing or incorrectly parses 843 sentences (285 + 558) and only succeeds in parsing 21 sentences (2.4%), whilst FARASA fails in parsing or incorrectly parses only 269 sentences (207 + 62) and succeeds in parsing 595 sentences (68.9%).

Farasa is then the parser tool we choose because it gives better results[7]. While analyzing these 595 sentences using Farasa, 52 different grammar rules have been extracted and, compared to other CNLs [8], it is enough to build a first version of the ACL. Table 3 presents an overview of these rules[8] with a code (used as rule reference) and the number of sentences for each rule. The used abbreviations, such as ج ف ,س ج, are explained in table 4.

TABLE 3: OVERVIEW OF THE ACL RULES WITH THE NUMBER OF SENTENCES PER RULE

Rule code	FARASA Parsing	Rule	# of sentences

[3] Extracted in October 3th, 2018 from: http://rowadaltamayoz.com/files/almanahij/11/Arabic1.pdf
[4] Extracted in October 28th, 2018 from: https://elearning.aljazeera.net/ar
[5] Extracted in November 2th, 2018 from: https://bialabanat.ahlamontada.net/
[6] Extracted in November 5th, 2018 from: https://analbahr.com/

[7] See Appendix B for all ACL rules using the FARASA syntactical output format.
[8] See Appendix A for the whole list of rules.

1.1	(TOP (S (NP (PRON)) (NP (NOUN) (ADJ)) (PUNC)))	ج س(مبتدأ(اسم إشارة)+ خبر(مفرد)+ نعت(صفة)).	19
6.1	(TOP (S (VP (V) (NP (DET+NOUN))) (PUNC)))	ج ف(فعل+ فاعل(اسم معرف بأل)).	31
6.8	(TOP (S (VP (V) (NP (NOUN))) (PUNC)))	ج ف(فعل+فاعل(اسم علم)).	18
7.1	(TOP (S (VP (V) (NP (DET+NOUN) (DET+NOUN))) (PUNC)))	ج ف(فعل+فاعل(اسم معرف بأل)+ مفعول به(اسم معرف بأل)).	27

TABLE 4: ABBREVIATIONS MEANING

Abbreviation	Meaning in Arabic	EnglishTranslation
ج س	جملة اسمية	Noun Phrase
ج ف	جملة فعلية	Verb Phrase
ش ج	شبه جملة	Phrase
ج اس	جملة استفهام	Interrogative sentence
ض من	ضمير منفصل	Detached - or independent pronoun
ض متر	ضمير متصل	attached pronoun
ض مس= ضمير	ضمير مستتر	hidden pronoun

To understand the structure of these rules, let us consider the rule 6.8:

(.((فعل+فاعل (اسم علم)) ج ف) / (TOP (S (VP (V) (NP (NOUN))) (PUNC)))

This rule means that the sentence is a verb phrase (ج ف) / (VP) and is made of a verb "فعل" / (V) followed by a subject "فاعل"/ (NOUN) as a proper noun or a named entity "أسم علم". For instance, the sentence " ذَهَبَ خالِدٌ." (Khaled has gone) follows this rule since the word "ذَهَبَ" (has gone) is a verb and is followed by "خالِدٌ" (Khaled) which is a proper noun/named entity.

As we can notice from Table 3, some rules are covered by more sentences than others. For instance, rules 6.1 or 7.1 are respectively covered with 31 and 27 sentences whilst the rule 1.1 is only covered by 19 sentences. This is due to the fact that these rules are the simplest ones for creating verbal sentences, that is why most of sentences belong to these rules. Figure 1 highlights the number of sentences per rule[9]. According to these statistics, the average number of sentences per rule is 10.6.

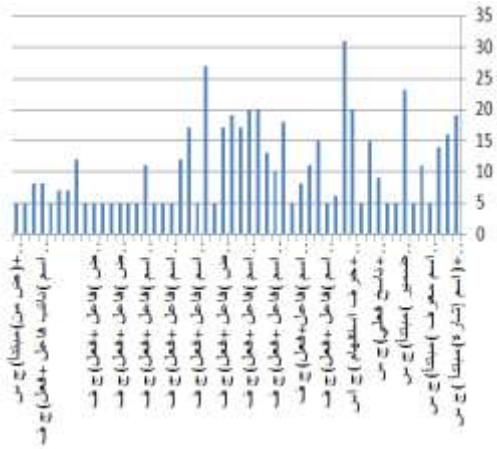


Figure 1. Number of sentences per rule

Now that the sentences are gathered and their corresponding rules are extracted, the next step consists in building the corpus using standard encoding to make it easily usable and sharable with interested researchers.

B. The encoding format of the ACL corpus

For standardization purposes, we choose the TEI standard encoding format [22]. Other formats could be interesting too, such as HTML, XML or Open Document Format but TEI remains the most used in building corpora.

TEI encoding format is an international project developed in 1988. It is also used as data storage and modeling systems [23]. TEI has been used to markup linguistic corpora such as GENIA corpus [24] and National Corpus of Polish [25]. Moreover, TEI is a well-designed and a widely accepted architecture [24]. Sperberget et al [26] present the main guidelines for TEI such as 1) the use in interchange between individuals and research groups using different programs and computer systems over a broad range of applications, 2)

[9] Rules covered with less than 5 sentences are rejected.

providing help to creating texts in electronic form, 3) the ability to be used for the local storage of text, 4) applicability to texts in any natural language, of any date, in any literary genre or text type, without restriction on form or content software packages requiring different input formats and 5) usefulness for the creation of electronic texts.

The structure of a TEI file is made of a header and a body. The header consists of a set of meta data that describe the corpus, such as the corpus name, the authors, the sources and further meta data. Figure 2 shows the header of our corpus.

Figure 2. Header of ACL Corpus

While the header is made of metadata, the body contains rules. Each rule has a code, a structure and all sentences respecting that rule. For each sentence, we store an id, the voyelled and unvowelled text as well as the result of parsing using Farasa. Figure 3 illustrates the body content of the rule 1.2.

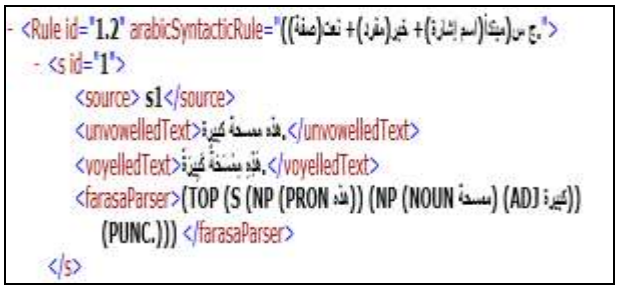


Figure 3. Body contentof the Rule1.2

These two steps have been applied to the whole 52 rules and 551 sentences and allowed creating an ACL corpus that contains grammar rules and sentences. In the next step, we provide a computing software we call ACL checker that checks whether a given Arabic sentence is accepted or not by our ACL.

III. THE ACL CHECKER

The main idea behind the ACL checker is to check whether an input sentence respects or not the ACL rules. To perform this, the checker parses the input sentence using Farasa Arabic parser, extracts the grammar rule, and then compares it with all ACL rules.

Figure 4 gives an overview of the ACL checker architecture. First, the user inputs a sentence that is passed to Farasa, which parses it and returns the corresponding syntactic tree (syntactic structures). The checker finally compares this structure with all ACL syntactic structures (see Appendix A). If the rule matches any of the existing rules, the sentence is then considered as covered by the ACL rules, otherwise the input sentence is rejected by the ACL checker.

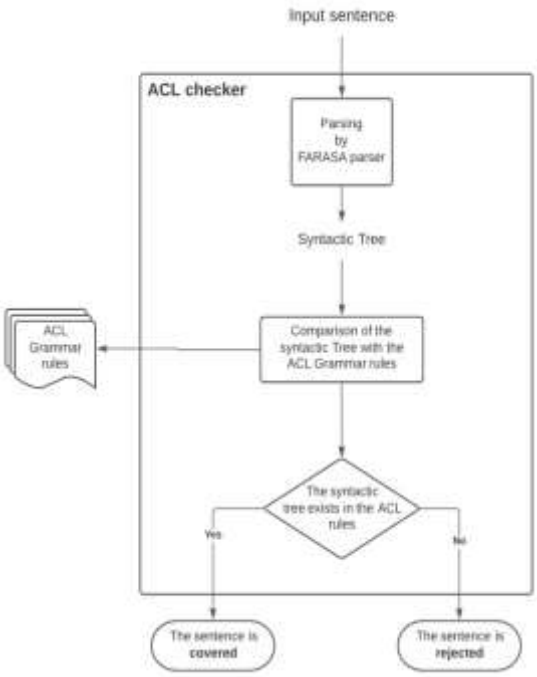


Figure 4. Overview of the ACL architecture

We show next the output of two executions of the ACL parser. In the first one, the system is given the sentence" هذه ممسحة كبيرة." translated in English to "This is a big mop.". As illustrated in Figure 5, the system starts by loading Farasa parser and loading the ACL rules. It calls then Farasa to parse the sentence and performs the ACL rules checking. For this first example, the sentence is covered by the ACL and it outputs the rule id, the rule and grammar structure.

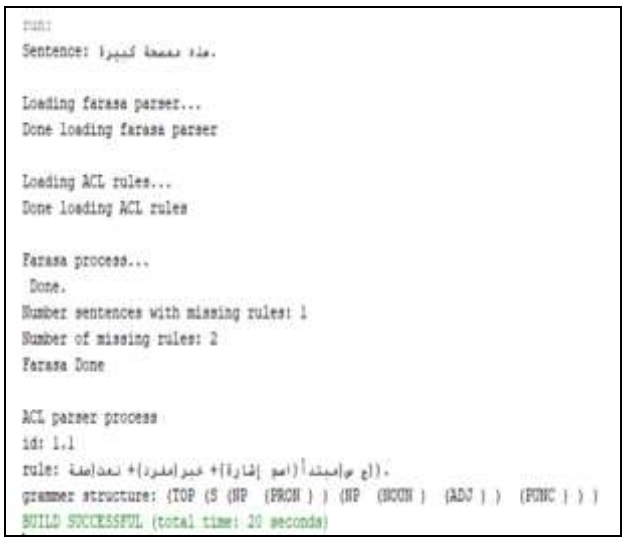


Figure 5. Analysis of a sentence covered by ACL

In the second example, we consider the sentence" هذا كوب كبير وجميل." translated in English to "This is a big and beautiful mug.". The same steps are executed by FARASA, but for this example, the rule extracted by FARASA does not match any ACL rule and then as illustrated in Figure 6, it outputs as message that the input sentence does not respect any ACL rule.

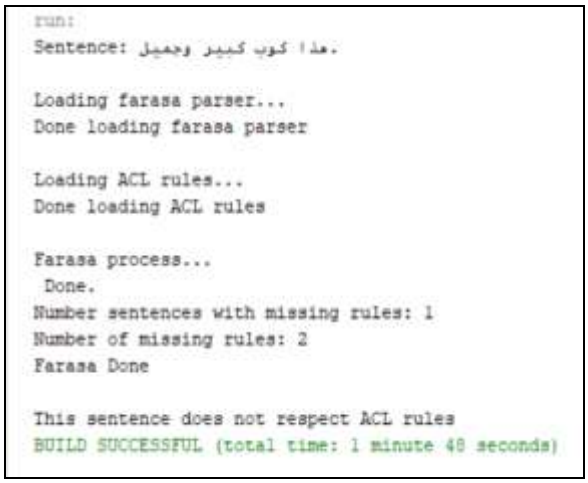


Figure 6. Analysis of a sentence not covered by ACL

IV. EXPERIMENTS

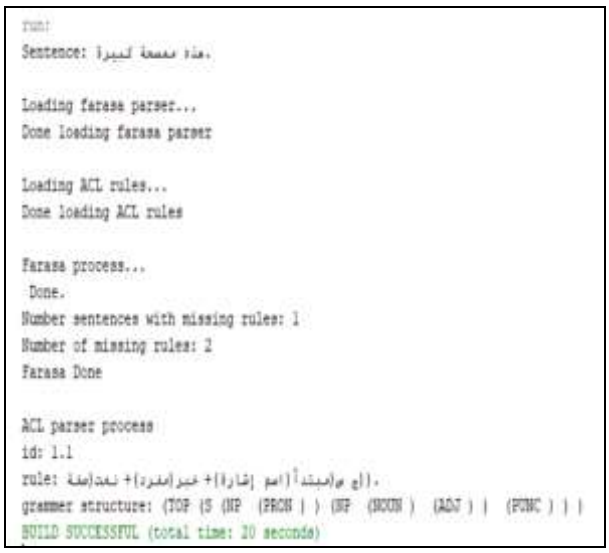
Three experiments are conducted concerning our ACL parser. The first one consists in the starting 551 sentences of our corpus. In the second, we focus on 30 incorrect sentences according to ACL rules, whilst in the third and last experiment, we use 200 correct sentences according to ACL rules, but with a different vocabulary than that of our corpus. Details about these experiments are provided in what follows.

Firstly, let's recall our reason for doing the first experiment It is noteworthy that the ACL parser does not use a statistical approach but is rather a purely rule-based tool. Indeed, the starting sentences were used to extract rules but the parser did not use them for learning or any other purpose. These sentences have been kept in ACL corpus and associated to their corresponding rules as rules' samples only. Obviously, we expect these 551 sentences to be correctly parsed by ACL parser. That is not to say that the parser already knows them, but because it knows the grammar rules that cover them. As result and as expected, all these sentences are successfully analyzed. Table 5 shows 3 samples of these sentences.

TABLE 5: SAMPLES OF SENTENCES USED IN THE FIRST EXPERIMENT

Sentence	**FARASA Parsing**	**ACL Rule**	**ACL Rule code**
هذه ممسحة كبيرة. (This is a big mop.)	(TOP (S (NP (PRON)) (NP (NOUN) (ADJ)) (PUNC)))	ج س(مبتدأ(اسم إشارة)+ خبر(مفرد)+ نعت(صفة)).	1.1
هذا قلم وذلك كتاب. (This is a pen and that is a book.)	(TOP (S (S (NP (PRON)) (NP (NOUN))) (CONJ) (S (NP (PRON)) (NP (NOUN)))	ج س(مبتدأ(اسم إشارة)+ خبر(اسم مفرد))+ حرف عطف+ ج س(مبتدأ(اسم إشارة)+ خبر(اسم مفرد)).	1.2
هذه التفاحة حمراء. (This apple is red)	(TOP (S (NP (NP (PRON)) (NP (DET+NOUN))) (NP (NOUN)) (PUNC.)))	ج س(مبتدأ (اسم إشارة) + بدل(اسم معرف بأل)+ خبر(اسم مفرد)).	1.3

Figure 7 illustrates the analysis performed by the ACL checker of the sentence "هذه ممسحة كبيرة." (translated in English to: This is a big mop). As illustrated, the checker starts with showing the input sentence. It then loads FARASA parser as well as the ACL rules. Next, the checker runs the parsing of the input sentence by FARASA, extracts its syntactic rule. Finally, and based on this rule, the ACL checker browses its rules and checks whether this rule is covered or not. In this case, the rule is covered and its details is displayed (id, rule in Arabic and grammar rule).



Figures 7. Analysis of the sentence "هذه ممسحة كبيرة." by ACL checker

It is worth mentioning that during the verification, 27% of the sentences give more than one rule. This is a natural behavior since these sentences are recognized in one hand by the corpus rules, but also with another rule although it was not included in the corpus. Table 6 shows samples of these sentences and figure 8 illustrates this behavior when analyzing the first sentence "نجح الطالب" translated to English as "The student succeeded".10

TABLE 6: SAMPLES OF SENTENCES THAT GIVE MORE THAN ONE SYNTACTIC RESULT

Sentence	**FARASA Parsing**	**ACL structure**	**ACL rule**	**ACL rule code**
نجح الطالب. (The student succeeded)	(TOP (S (VP (V نجح) (NP (DET+NOUN الطالب))) (PUNC.)))	(TOP (S (VP (V) (NP (DET+NOUN))) (PUNC.)))	ج ف(فعل+ فاعل(اسم معرف بأل)).	6.1
			ج ف(فعل+ نائب الفاعل(اسم معرف بأل)).	10.2
ينجز العمال العمل متعاونين. (The workers do the work cooperatively.)	(TOP (S (VP (V ينجز) (NP (DET+NOUN العمال) (ADJ متعاونين))) (PUNC.)))	(TOP (S (VP (V) (NP (DET+NOUN) (NP (ADJ))) (PUNC.)))	ج ف(فعل+ فاعل(اسم معرف بأل)+ مفعول به1(اسم معرف بأل)+ مفعول به2(اسم نكرة)).	8.1
جاء زيد نفسه. (Zaid himself came.)	(TOP (S (VP (V جاء) (NP (NOUN زيد) (NP (NOUN نفس) (NP (PRON ه))))) (PUNC.)))	(TOP (S (VP (V) (NP (NOUN) (NP (NOUN) (NP (PRON)))) (PUNC.)))	ج ف(فعل+ فاعل(اسم علم)+ توكيد مضاف+ مضاف إليه(ضمت)).	6.11
			ج ف(فعل+ فاعل(اسم علم)+ مفعول به مضاف+ مضاف إليه(ضمت)).	7.8

As we notice in Figure 8, FARASA parser extracts 2 syntactic rules, that the ACL checker maps to two ACL rules. At this stage, the checker does not choose between them and outputs both of them as a result.

[10] Appendix C lists all ACL rules that can be confusing

Figure 8. The syntactic parsing of the sentence "نجح الطالب"

In the second experiment, we focus on 30 incorrect ACL sentences. This allows to ensure that the checker accepts only ACL rules and rejects the others. Table 7 gives examples of those sentences alongside their non-ACL rules. The result is that none of them are accepted by the checker.

TABLE 7: EXAMPLES OF SENTENCES USED IN THE SECOND EXPERIMENT

Sentence	FARASA Parsing
هذا كوب كبير وجميل. (This is a big and beautiful mug.)	(TOP (S (NP (PRON)) (NP (NOUN) (NP (ADJ) (CONJ) (NOUN))) (PUNC)))
أكل الهر اللحم في المنزل. (The cat ate meat at home.)	(TOP (S (VP (V) (NP (DET+NOUN) (DET+NOUN)) (PP (PREP) (NP (DET+NOUN))))) (PUNC)))
ذهبت سمر بالسيارة الحمراء. (Samar went in the red car.)	(TOP (S (VP (V+PRON) (VP (V) (PP (PREP) (NP (DET+NOUN) (DET+NOUN)))))) (PUNC)))

Figure 9 shows the analysis of one not covered sentence. As we notice, Farasa performs its analysis, and since the returned rule is not covered by ACL rules, this sentence is rejected and considered as not respecting ACL rules.

```
run:
Sentence: هذا كوب كبير وجميل.

Loading farasa parser...
Done loading farasa parser

Loading ACL rules...
Done loading ACL rules

Farasa process...
 Done.
Number sentences with missing rules: 1
Number of missing rules: 2
Farasa Done

This sentence does not respect ACL rules
BUILD SUCCESSFUL (total time: 1 minute 48 seconds)
```

Figure 9. The syntactic parsing of the sentence " هذا كوب كبير وجميل."

In the third and final experiment, we use 200 sentences which are, from a linguistic point of view, covered by ACL rules. The only difference is that these sentences use a vocabulary not covered by the corpus. Table 8 shows some examples of this kind of sentences.

TABLE 8: EXAMPLES OF SENTENCES USED IN THE THIRD EXPERIMENT

Sentence	FARASA Parsing	ACL Rule	ACL Rule Code
هذا مفتاح كبير. (This is a big key)	(TOP (S (NP (PRON)) (NP (NOUN) (ADJ)) (PUNC)))	ج س(مبتدأ(اسم إشارة)+ خبر(مفرد)+ نعت(صفة)).	1.1
هذا برتقال وذلك عنب. (This is an orange and that is a grape)	(TOP (S (S (NP (PRON)) (NP (NOUN))) (CONJ) (S (NP (PRON)) (NP (NOUN))) (PUNC)))	ج س(مبتدأ(اسم إشارة)+ خبر(اسم مفرد))+ حرف عطف+ ج س(مبتدأ(اسم إشارة)+ خبر(اسم مفرد)).	1.2
هذه اليمونة خضراء. (This lemon is green)	(TOP (S (NP (NP (PRON)) (NP (DET+NOUN))) (ADJP (ADJ)) (PUNC)))	ج س(مبتدأ (اسم إشارة) + بدل(اسم معرف بأل)+ خبر(اسم مفرد)).	1.3

In this last experiment, the parser succeeds in parsing 197 out of 200 and fails in only 3. Although 3 is a low rate (1.5%), its analysis shows that the problem concerns some confusing vocabulary. For example, the analysis by FARASA of the sentence "ذهبت سمر بالسيارة" translated in English to "Samar left by car" gives the rule (Verb-Verb-PREP-Noun) instead of (Verb-Noun-PREP-Noun) due to the confusing word "سمر" (Samar). Indeed, this word has various meanings; it means a first name of a girl (Samar) or the action to stay awake late at night talking to someone. At the end, although this example is correct in Arabic and has a clear and unambiguous meaning, its syntactic interpretation as (Verb-Verb- PREP-Noun) is not covered by ACL. Consequently, a disambiguation process based on semantic constraints is necessary to eliminate such interpretations.

TABLE9.SUMMARY OF THE EXPERIMENTS

Experiment	# of sentences	Result a expected	%
Sentences of the corpus	551	551	100 %
Incorrect sentences	30	30	100%
Correct sentence with unknown vocabulary	200	197	98.5 %
Summary	781	778	99.6%

As a summary, table 9 shows the result of these three experiments conducted using ACL checker. The ACL parser is assessed on different types of sentences; initially, on the

551 sentences of the corpus, then on 30 incorrect sentences and finally on 200 sentences whose vocabulary is different from that of the ACL corpus. The bottom line is that 99.6% of these experiments were successful, which is good enough and satisfactory for the first version of ACL.

V. CONCLUSIONS AND FUTURE WORKS

This paper introduced the construction of the Arabic Controlled Language named ACL. In our approach, we started by building a corpus composed of grammar rules, their syntactic structures and samples. These rules have been taken from textbooks and websites dedicated to teaching Arabic language to kids. This ended up with a corpus of 52 grammar rules and 551 Arabic sentences. For standardization purposes and for better reusability, this corpus was formulated using the standard TEI encoding format.

Furthermore, we provided a tool called ACL checker which allows to check sentences according to the ACL rules. This tool is 100% rule-based and doesn't use statistical models or machine learning techniques. Three experiments have been carried out to assess its efficiency and accuracy and the results were satisfactory for a first version of ACL and its checker.

As future works, we plan to:

- add a semantic component to ACL to extract the meaning from sentences and formulates it using one of the knowledge representation formalisms such as Conceptual Graphs [27][28] or RDF [29][30].
- integrate this work into Amine Platform [31][32] and SAFAR framework [33] to be able to use it in conjunction with other complementary modules they both provide such as semantic reasoning, information retrieval or machine translation;
- add more rules allowing to cover a wider part of Arabic and thereby be useful in advanced NLP systems.

REFERENCES

[1] Schwitter, R. (2010, August). Controlled natural languages for knowledge representation. In Coling 2010: Posters (pp. 1113-1121).

[2] Schwitter, R. (2005, December). Controlled Natural Language as Interface Language to the Semantic Web. In IICAI (pp. 1699-1718).

[3] Hayes, P., Maxwell, S., &Schmandt, L. (1996, March). Controlled English advantages for translated and original English documents. In Proceedings of CLAW (pp. 84-92).

[4] Kuhn, T. (2014). A survey and classification of controlled natural languages. Computational linguistics, 40(1), 121-170.

[5] Haralambous, Y., Sauvage-Vincent, J., & Puentes, J. (2014, August). INAUT, a controlled language for the French coast pilot books instructions nautiques. In International Workshop on Controlled Natural Language (pp. 102-111). Springer, Cham.

[6] Höfler, S., &Bünzli, A. (2010). Designing a controlled natural language for the representation of legal norms. Bernth, Arendse. "EasyEnglish: a tool for improving document quality." Proceedings of the fifth conference on Applied natural language processing. Association for ComputationalLinguistics, 1997.

[7] Elazhary, H. (2016). CAL: A controlled Arabic language for authoring ontologies. Arabian Journal for Science and Engineering, 41(8), 2911-2926.

[8] El Fahal, H. S., Nasri, M., Bouzoubaa, K., &Kabbaj, A. (2019). Roadmap for an Arabic Controlled Language. International Journal of Information Technology and Language Studies, 3(3).

[9] Voice of America. 2009. VOA Special English Word Book: A List of Words Used in Special English Programs on Radio, Television, and the Internet,Washington, DC. Warren, David H. D. and Fernando

[10] SEC (Securities and Exchange Commission). "A plain English handbook: How to create clear SEC disclosure documents." US Securities and Exchange Commission, Washington, DC (1998).

[11] Authority, C. A. (2010). CAP 413: Radiotelephony Manual.

[12] Rychtyckyj, N. (2005). Ergonomics analysis for vehicle assembly using artificial intelligence. AI Magazine, 26(3), 41-41.

[13] Bernth, A. (1997, March). EasyEnglish: a tool for improving document quality. In Fifth Conference on Applied Natural Language Processing (pp. 159-165).

[14] Fuchs, N. E., &Schwitter, R. (1996). Attempto controlled english (ace). arXiv preprint cmp-lg/9603003.

[15] White, C., &Schwitter, R. (2009, December). An update on PENG light. In Proceedings of the

Australasian Language Technology Association Workshop 2009 (pp. 80-88).

[16] Van Renssen, A. S. H. P. (2005). Gellish: a generic extensible ontological language-design and application of a universal data structure.

[17] Van Renssen, A. (2013). Gellish Formal English. Definition and Application of a Universal Information Modeling Language.

[18] Huijsen, W. (1998). Introduction to controlled languages.

[19] Shaalan, K. (2014). A survey of arabic named entity recognition and classification. Computational Linguistics, 40(2), 469-510.

[20] Klein, D., & Manning, C. D. (2003, July). Accurate unlexicalized parsing. In Proceedings of the 41st annual meeting of the association for computational linguistics (pp. 423-430).

[21] Abdelali, A., Darwish, K., Durrani, N., & Mubarak, H. (2016, June). Farasa: A fast and furious segmenter for arabic. In Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Demonstrations (pp. 11-16).

[22] Ide, N. (1994). Encoding standards for large text resources: The Text Encoding Initiative. In COLING 1994 Volume 1: The 15th International Conference on Computational Linguistics.

[23] Centre for Information Modelling,"What is text, really? TEI and beyond Book of Abstracts.",September 16–20, 2019.

[24] Erjavec, T., Kim, J. D., Ohta, T., Tateisi, Y., &Tsujii, J. I. (2003, July). Encoding biomedical resources in TEI: the case of the GENIA Corpus. In Proceedings of the ACL 2003 workshop on natural language processing in biomedicine (pp. 97-104).

[25] Bański, P., &Przepiórkowski, A. (2009, August). Stand-off TEI annotation: the case of the National Corpus of Polish. In Proceedings of the Third Linguistic Annotation Workshop (LAW III) (pp. 64-67).

[26] Sperberg-McQueen, C. M., & Burnard, L. (Eds.). (1994). Guidelines for electronic text encoding and interchange (Vol. 1). Chicago and Oxford: Text Encoding Initiative.

[27] Sowa, J. F. (1984). Conceptual structures: information processing in mind and machine. Addison-Wesley Longman Publishing Co., Inc..

[28] Sowa, J. F. (2008). Conceptual Graphs. In van Harmelen, F., Lifschitz, V., & Porter, B.(eds) Handbook of Knowledge Representation.

[29] Lassila, O., & Swick, R. R. (1998). Resource description framework (RDF) model and syntax specification.

[30] Lassila, O. (1999). Resource description framework (RDF) model and syntax specification, W3C recommendation. http://www. w3. org/TR/PR-rdf-syntax.

[31] Kabbaj, A., Bouzouba, K., &Soudi, A. (2005). Amine platform: an artificial intelligence environment for the development of intelligent systems. Artificial Intelligence.

[32] Kabbaj, A. (2006, July). Development of intelligent systems and multi-agents systems with amine platform. In International Conference on Conceptual Structures (pp. 286-299). Springer, Berlin, Heidelberg.

[33] Jaafar Younes and Bouzoubaa Karim. 2017, SAFAR: Software Architecture For Arabic language pRocessing, Available: http://arabic.emi.ac.ma/safar/. [Accessed 1 10 2018].

Appendix

Appendix A: ACL rules with their syntactic structures

Rule code	Syntactic structure
1.1	(TOP (S (NP (PRON)) (NP (NOUN) (ADJ)) (PUNC)))
	(TOP (S (NP (PRON)) (NP (NOUN) (NOUN)) (PUNC)))
1.2	(TOP (S (S (NP (PRON)) (NP (NOUN))) (CONJ) (S (NP (PRON)) (NP (NOUN))) (PUNC)))
	(TOP (S (S (NP (PRON)) (ADJP (ADJ))) (CONJ) (S (NP (PRON)) (NP (NOUN))) (PUNC)))
	(TOP (S (S (NP (PRON)) (VP (V))) (CONJ) (S (NP (PRON)) (NP (NOUN))) (PUNC)))
1.3	(TOP (S (NP (NP (PRON)) (NP

	(DET+NOUN))) (ADJP (ADJ)) (PUNC)))
	(TOP (S (NP (NP (PRON)) (NP (DET+NOUN))) (NP (NOUN)) (PUNC)))
2.1	(TOP (S (NP (DET+NOUN)) (SBAR (WHNP (PART)) (S (VP (V+PRON) (NP (DET+NOUN))))) (PUNC)))
3.1	(TOP (S (NP (PRON)) (VP (V+PRON)) (PUNC)))
	(TOP (S (NP (PRON)) (VP (V) (NP (PRON))) (PUNC)))
	(TOP (S (NP (PRON)) (VP (V) (NP (DET+NOUN) (DET+ADJ))) (PUNC)))
3.2	(TOP (S (NP (PRON)) (NP (NOUN) (NP (DET+NOUN) (DET+ADJ))) (PUNC)))
3.3	(TOP (S (NP (PRON)) (VP (V)) (PUNC)))
3.4	(TOP (S (NP (PRON)) (VP (V) (NP (DET+NOUN))) (PUNC)))
	(TOP (S (NP (PRON)) (VP (V) (NP (PRON)) (NP (DET+NOUN))) (PUNC)))
	(TOP (S (S (NP (PRON)) (VP (V) (NP (DET+NOUN)))) (PUNC)))
3.5	(TOP (S (NP (PRON)) (VP (V) (NP (PRON)) (NP (NOUN) (NP (PRON)))) (PUNC)))
3.6	(TOP (S (S (NP (PRON)) (VP (V))) (CONJ) (S (VP (V) (NP (DET+NOUN)))) (PUNC)))
4.1	(TOP (S (VP (V) (ADJP (DET+NOUN) (NOUN))) (PUNC)))
	(TOP (S (VP (V) (NP (NP (DET+NOUN)) (NOUN))) (PUNC)))
4.2	(TOP (S (VP (V) (NP (NOUN) (NOUN))) (PUNC)))
4.3	(TOP (S (VP (V) (NP (DET+NOUN)) (NP (NOUN) (ADJ))) (PUNC)))
	(TOP (S (VP (V) (NP (DET+NOUN)) (NP (NOUN) (NOUN))) (PUNC)))
5.1	(TOP (SQ (PRT (PART)) (VP (V) (NP (NP (NOUN) (DET+NOUN)) (NP (NOUN)))) (PUNC)))
	(TOP (SQ (PRT (PART)) (VP (V) (NP (NOUN) (DET+NOUN))) (PUNC)))
6.1	(TOP (S (VP (V) (NP (DET+NOUN))) (PUNC)))
6.2	(TOP (S (VP (V) (NP (NP (DET+NOUN)) (PP (PREP) (NP (DET+NOUN))))) (PUNC)))
	(TOP (S (VP (V) (NP (NP (DET+NOUN)) (NP (NOUN) (NP (DET+NOUN))))) (PUNC)))
	(TOP (S (VP (V) (NP (DET+NOUN)) (PP

	(PREP) (NP (DET+NOUN)))) (PUNC)))
	(TOP (S (VP (V) (NP (DET+ADJ)) (PP (PREP) (NP (DET+NOUN)))) (PUNC)))
6.3	(TOP (S (VP (V) (NP (DET+NOUN)) (PP (PREP) (NP (DET+NOUN) (CONJ) (DET+NOUN)))) (PUNC)))
6.4	(TOP (S (VP (V) (NP (DET+NOUN) (NOUN))) (PUNC)))
	(TOP (S (VP (V) (NP (DET+NOUN) (ADJ))) (PUNC)))
	(TOP (S (VP (V) (NP (DET+NOUN)) (NP (ADJ))) (PUNC)))
	(TOP (S (VP (V) (NP (DET+NOUN)) (NP (NOUN))) (PUNC)))
6.5	(TOP (S (VP (V) (NP (DET+NOUN) (DET+NOUN))) (PUNC)))
	(TOP (S (VP (V) (NP (DET+NOUN) (DET+ADJ))) (PUNC)))
6.6	(TOP (S (VP (V) (NP (NP (DET+NOUN)) (SBAR (WHNP (PART)) (S (V))))) (PUNC)))
	(TOP (S (VP (V) (NP (NP (DET+NOUN)) (SBAR (WHNP (PART)) (S (NOUN)))))) (PUNC)))
6.7	(TOP (S (VP (V) (NP (DET+NOUN) (NOUN))) (PUNC)))
	(TOP (S (VP (V) (NP (NP (DET+NOUN)) (NP (NOUN)))) (PUNC)))
6.8	(TOP (S (VP (V) (NP (NOUN))) (PUNC)))
6.9	(TOP (S (VP (V+PRON) (NP (NOUN)) (PP (PREP) (NP (DET+NOUN))))) (PUNC)))
	(TOP (S (VP (V+PRON) (NP (NP (ADJ)) (PP (PREP) (NP (DET+NOUN)))))) (PUNC)))
	(TOP (S (VP (V+PRON) (NP (NP (NOUN)) (PP (PREP) (NP (DET+NOUN))))) (PUNC)))
	(TOP (S (VP (V+PRON) (ADJP (ADJ) (PP (PREP) (NP (DET+NOUN))))) (PUNC)))
	(TOP (S (VP (V+PRON) (NP (ADJ)) (PP (PREP) (NP (DET+NOUN)))) (PUNC)))
6.10	(TOP (S (VP (V) (NP (NOUN) (ADJ))) (PUNC)))
6.11	(TOP (S (VP (V) (NP (NOUN) (NP (NOUN) (NP (PRON))))) (PUNC)))
6.12	(TOP (S (VP (V) (NP (NOUN) (NP (DET+NOUN)))) (PUNC)))
	(TOP (S (VP (V) (NP (NOUN) (DET+NOUN))) (PUNC)))

6.13	(TOP (S (VP (V) (NP (NOUN) (CONJ) (NOUN))) (PUNC)))
6.14	(TOP (S (VP (V+PRON) (PP (PREP) (NP (NOUN) (NP (DET+NOUN)))))) (PUNC)))
	(TOP (S (VP (V+PRON) (PP (PREP) (NP (NOUN) (DET+NOUN)))) (PUNC)))
6.15	(TOP (S (VP (V+PRON) (PP (PREP) (NP (NOUN) (NP (NOUN) (NP (PRON))))))) (PUNC)))
	(TOP (S (VP (V+PRON) (NP (NOUN) (NP (NOUN) (NP (PRON))))) (PUNC)))
6.16	(TOP (S (VP (PRT (PART)) (V) (PP (PREP) (NP (DET+NOUN)))) (PUNC)))
7.1	(TOP (S (VP (V) (NP (DET+NOUN) (DET+NOUN))) (PUNC)))
7.2	(TOP (S (VP (V) (NP (DET+ADJ) (NOUN))) (PUNC)))
7.3	(TOP (S (VP (V) (NP (DET+NOUN)) (NP (NOUN) (NP (PRON)))) (PUNC)))
	(TOP (S (VP (V+PRON) (NP (DET+NOUN)) (NP (NOUN) (NP (PRON)))) (PUNC)))
7.4	(TOP (S (VP (V) (NP (DET+NOUN)) (NP (NOUN) (NP (PRON)))) (PUNC)))
7.5	(TOP (S (VP (V+PRON) (NP (NP (DET+NOUN) (DET+NOUN)) (SBAR (WHNP (PART)) (S (VP (V+PRON) (NP (NOUN) (NP (PRON)))))))) (PUNC)))
	(TOP (S (VP (V) (NP (NP (DET+NOUN) (DET+NOUN)) (SBAR (WHNP (PART)) (S (VP (V+PRON) (NP (NOUN) (NP (PRON)))))))) (PUNC)))
7.6	(TOP (S (VP (V) (NP (DET+NOUN)) (NP (NP (DET+NOUN) (DET+ADJ)) (SBAR (VP (V) (NP (NOUN)) (PP (PREP) (NP (DET+NOUN))))))) (PUNC)))
	(TOP (S (VP (V) (NP (NP (DET+NOUN) (DET+NOUN) (DET+ADJ)) (SBAR (VP (V) (NP (NOUN)) (PP (PREP) (NP (DET+NOUN))))))) (PUNC)))
7.7	(TOP (S (VP (V) (NP (NP (DET+NOUN) (DET+NOUN)) (SBAR (WHNP (PART)) (S (VP (V) (NP (PRON))))))) (PUNC)))
	(TOP (S (VP (V) (NP (NP (DET+NOUN) (DET+NOUN)) (SBAR (WHNP (PART)) (S (VP (V) (NP (PRON))))))) (PUNC)))
7.8	(TOP (S (VP (V) (NP (NOUN) (NP (NOUN) (NP (PRON))))) (PUNC)))
	(TOP (S (VP (V) (NP (NOUN) (NOUN) (NP (PRON)))) (PUNC)))
7.9	(TOP (S (VP (V+PRON) (NP (DET+NOUN))) (PUNC)))
	(TOP (S (VP (V) (NP (PRON)) (NP (DET+NOUN))) (PUNC)))
7.10	(TOP (S (VP (V+PRON) (NP (DET+NOUN) (DET+ADJ))) (PUNC)))
	(TOP (S (VP (V) (NP (PRON)) (NP (DET+NOUN) (DET+ADJ))) (PUNC)))
7.11	(TOP (S (VP (V+PRON) (NP (DET+NOUN) (NOUN))) (PUNC)))
7.12	(TOP (S (VP (V+PRON) (NP (NOUN) (NP (DET+NOUN)))) (PUNC)))
7.13	(TOP (S (VP (V+PRON) (NP (NOUN) (NP (NOUN) (NP (PRON))))) (PUNC)))
7.14	(TOP (S (VP (V+PRON) (NP (NOUN) (ADJ))) (PUNC)))
7.15	(TOP (S (VP (V) (NP (NP (DET+NOUN)) (SBAR (WHNP (PART)) (S (VP (V) (NP (DET+NOUN)) (PP (PREP) (NP (NOUN)))))))) (PUNC)))
7.16	(TOP (S (VP (V) (NP (PRON)) (NP (NOUN) (NP (PRON)))) (PUNC)))
8.1	(TOP (S (VP (V) (NP (DET+NOUN) (DET+NOUN) (ADJ))) (PUNC)))
	(TOP (S (VP (V) (NP (DET+NOUN) (DET+NOUN) (NOUN))) (PUNC)))
	(TOP (S (VP (V) (NP (NOUN) (NP (DET+NOUN) (NOUN)))) (PUNC)))
	(TOP (S (VP (V) (NP (NOUN)) (S (NP (DET+NOUN)) (ADJP (ADJ)))) (PUNC)))
	(TOP (S (VP (V) (NP (DET+NOUN) (DET+NOUN)) (NP (NOUN))) (PUNC)))
8.2	(TOP (S (VP (V) (NP (NOUN) (DET+NOUN)) (NP (NOUN))) (PUNC)))
	TOP (S (VP (V) (NP (NOUN) (DET+NOUN) (NOUN))) (PUNC)))
9.1	(TOP (S (VP (PRT (PART)) (V) (NP (NP (DET+NOUN)) (CONJ) (DET+NOUN))) (PUNC)))
10.1	(TOP (S (VP (V) (NP (NOUN))) (PUNC)))
10.2	(TOP (S (VP (V) (NP (DET+NOUN))) (PUNC)))
11.1	(TOP (S (S (VP (V) (NP (DET+NOUN)))) (CONJ) (S (VP (V+PRON) (NP (DET+NOUN)))) (PUNC)))

Appendix B: ACL Rules with the number of sentences per rule

Rule code	Rule	# of sentences
1.1	ج س(مبتدأ(اسم إشارة)+ خبر(مفرد)+ نعت(صفة)).	19
1.2	ج س(مبتدأ(اسم إشارة)+ خبر(اسم مفرد))+ حرف عطف+ ج س(مبتدأ(اسم إشارة)+ خبر(اسم مفرد)).	16
1.3	ج س(مبتدأ (اسم إشارة) + بدل(اسم معرف بأل)+ خبر(اسم مفرد)).	14
2.1	ج س(مبتدأ(اسم معرف بأل)+خبر(جملة فعلية(حرف نفي+ فعل+ فاعل(ض مس)+ مفعول به(اسم معرف بأل))))).	5
3.1	ج س(مبتدأ(ضمير منفصل)+ خبر(ج فعلية(فعل+ فاعل(ض مت))))).	11
3.2	ج س(مبتدأ(ضمير منفصل)+ خبر مضاف+ مضاف إليه(اسم معرف بأل)+ نعت(اسم معرف بأل)).	5
3.3	ج س(مبتدأ(ضمير منفصل)+ خبر(ج فعلية(فعل+ فاعل(ض مس))))).	23
3.4	ج س(مبتدأ(ضمير منفصل)+ خبر(ج فعلية(فعل+ مفعول به(ض مت)+ فاعل(اسم معرف بأل))))).	5
3.5	ج س(مبتدأ(ضمير منفصل)+ خبر(ج فعلية(فعل+ فاعل(ض مت)+ مفعول به مضاف+ مضاف إليه(ض مت))))).	5
3.6	ج س(مبتدأ(ض من)+ خبر(ج فعلية(فعل+ فاعل(ض مس)+ حرف عطف+ (ج فعلية(فعل+ فاعل(ض مس)+ مفعول به1(اسم معرف بأل)))))).	5
4.1	ج س(ناسخ فعلي+ اسمها(اسم معرف بأل) + خبرها(اسم مفرد)).	9
4.2	ج س(ناسخ فعلي+ اسمها(اسم علم)+ خبرها(اسم مفرد)).	15
4.3	ج س(ناسخ فعلي+ اسمها(اسم معرف بأل)+ خبرها(اسم مفرد)+ نعت).	5
5.1	ج اس(حرف استفهام+ فعل+ فاعل(اسم علم)+ مفعول به(اسم معرف بأل))؟	20
6.1	ج ف(فعل+ فاعل(اسم معرف بأل)).	31
6.2	ج ف(فعل+فاعل(اسم معرف بأل) +حرف جر+ اسم مجرور(اسم معرف بأل)).	6
6.3	ج ف(فعل+ فاعل(اسم معرف بأل)+ حرف جر+ اسم مجرور(اسم معرف بأل)+ حرف عطف+ اسم معطوف(اسم معرف بأل)).	5
6.4	ج ف(فعل+ فاعل(اسم معرف بأل)+ حال(مفرد)).	15
6.5	ج ف(فعل+ فاعل(اسم معرف بأل)+ نعت(معرف بأل)).	11
6.6	ج ف(فعل+فاعل(اسم معرف بأل)+ نعت(اسم موصول)+ جملة الصلة(ج فعلية(فعل+ فاعل(ض مس))))).	8
6.7	ج ف(فعل+ فاعل(اسم معرف بأل)+ مفعول فيه(ظرف زمان)).	5
6.8	ج ف(فعل+فاعل(اسم علم)).	18
6.9	ج ف(فعل+ فاعل(اسم علم)+ حرف جر+ اسم مجرور(اسم معرف بأل)).	10
6.10	ج ف(فعل+ فاعل(اسم علم)+ حال(مفرد)).	13
6.11	ج ف(فعل+ فاعل(اسم علم)+ توكيد مضاف+ مضاف إليه(ض مت)).	20
6.12	ج ف(فعل+ فاعل(اسم علم)+ نعت(اسم معرف بأل)).	20
6.13	ج ف(فعل +فاعل(اسم علم)+حرف عطف+ اسم معطوف(اسم علم)).	17
6.14	ج ف(فعل+ فاعل(ض مت)+ حرف جر+ اسم مجرور(اسم علم)+ نعت(اسم معرف بأل)).	19
6.15	ج ف(فعل+ فاعل(ض مت)+ حرف جر+ اسم مجرور(اسم علم)+ توكيد مضاف+ مضاف إليه(ض مت)).	17
6.16	ج ف(حرف نفي+ فعل+ فاعل(ض مس)+ حرف جر+ اسم مجرور(اسم معرف بأل)).	5
7.1	ج ف(فعل+فاعل(اسم معرف بأل)+ مفعول به(اسم معرف بأل)).	27
7.2	ج ف(فعل+ فاعل(اسم معرف بأل)+ مفعول به(اسم نكرة)).	17
7.3	ج ف(فعل+ فاعل(اسم معرف بأل)+ مفعول به مضاف+ مضاف إليه(ض مت)).	12
7.4	ج ف(فعل+ فاعل(اسم معرف بأل)+ مفعول به(اسم معرف بأل)+ حال(مفرد)).	5
7.5	ج ف(فعل+ فاعل(اسم معرف بأل)+ مفعول به(اسم معرف بأل)+ نعت(اسم موصول)+ جملة الصلة(ج فعلية(فعل+ فاعل(ض مت)+ مفعول به مضاف+ مضلف إليه(ض مت))))).	5
7.6	ج ف(فعل+ فاعل(اسم معرف بأل)+ مفعول به(اسم معرف بأل)+ نعت(اسم موصول)+ جملة الصلة(ج فعلية(فعل+ فاعل(ض مت)+ حرف جر+ اسم مجرور(اسم معرف بأل))))).	5
7.7	ج ف(فعل+ فاعل(اسم معرف بأل)+ مفعول به(اسم معرف بأل)+ نعت(اسم موصول)+ جملة الصلة(ج فعلية(فعل+ فاعل(ض مس)+ مفعول به(ض مت))))).	5
7.8	ج ف(فعل+ فاعل(اسم علم)+ مفعول به مضاف+ مضاف إليه(ض مت)).	11
7.9	ج ف(فعل+ فاعل(ض مت)+ مفعول به(اسم معرف بأل)).	5
7.1	ج ف(فعل+ فاعل(ض مت)+ مفعول به(اسم معرف بأل)+ نعت(اسم معرف بأل)).	5
7.11	ج ف(فعل+ فاعل(ض مت)+ مفعول به(اسم معرف بأل)+ حال(مفرد)).	5
7.12	ج ف(فعل + فاعل(ض مت)+ مفعول به(اسم علم)+ نعت(اسم معرف بأل)).	5
7.13	ج ف(فعل+ فاعل(ض مت)+ مفعول به(اسم علم)+ توكيد مضاف+ مضاف إليه(ض مت)).	5
7.14	ج ف(فعل+ فاعل(ض مت)+ مفعول به(اسم نكرة)+ نعت(صفة)).	5
7.15	ج ف(فعل+ فاعل(ض مس)+ مفعول به(اسم	5

	معرف بأل)+ نعت(اسم موصول)+ جملة الصلة(ج فعلية(فعل+ فاعل(ض مس)+ مفعول به(اسم معرف بأل)+ حال(ش ج(حرف جر+ اسم مجرور(اسم نكرة))))))).	
7.16	ج ف(فعل+ مفعول به(ض مت)+ فاعل مضاف+ مضاف إليه(ض مت)).	12
8.1	ج ف(فعل+ فاعل(اسم معرف بأل)+ مفعول به1(اسم معرف بأل)+ مفعول به2(اسم نكرة)).	7
8.2	ج ف(فعل+ فاعل(اسم علم)+ مفعول به1(اسم معرف بأل)+ مفعول به2(اسم نكرة)).	7
9.1	ج ف(حرف نهي وجزم+ فعل+فاعل(ض مس)+ مفعول به(اسم معرف بأل)+ حرف عطف+ اسم معطوف(اسم معرف بأل)).	5
10.1	ج ف(فعل+ نائب الفاعل(اسم علم)).	8
10.2	ج ف(فعل+ نائب الفاعل(اسم معرف بأل)).	8
11.1	ج ف(فعل+ فاعل(اسم معرف بأل))+ حرف عطف+ (ج فعلية(فعل+ فاعل(ض مت)+ مفعول به(اسم معرف بأل))))).	5

Appendix C: ACL rules corresponding to the same syntactic structure

Rules codes	**Syntactic structure**
6.11,7.8	(TOP (S (VP (V) (NP (NOUN) (NP (NOUN) (NP (PRON))))) (PUNC)))
7.4,8.1	(TOP (S (VP (V) (NP (DET+NOUN) (DET+NOUN) (ADJ))) (PUNC)))
6.8,10.1	(TOP (S (VP (V) (NP (NOUN))) (PUNC)))
6.1,10.2	(TOP (S (VP (V) (NP (DET+NOUN))) (PUNC)))
6.4,6.7	(TOP (S (VP (V) (NP (DET+NOUN) (NOUN))) (PUNC)))
6.5,7.1	(TOP (S (VP (V) (NP (DET+NOUN) (DET+NOUN))) (PUNC)))
7.13,6.15	(TOP (S (VP (V+PRON) (NP (NOUN) (NP (NOUN) (NP (PRON))))) (PUNC)))