Parsing Arabic Nominal Sentences with Transducers to Annotate Corpora

Nadia Ghezaiel Hammouda[1], Kais Haddar[2]

[1] Higher Institute of Computer and Communication Technologies of Hammam Sousse,
Miracl Laboratory, Tunisia

[2] University of Sfax, Faculty of Sciences of Sfax, Miracl Laboratory,
Tunisia

ghezaielnadia.ing@gmail.com, Kais.haddar@yahoo.fr

Abstract. Studying Arabic nominal sentences is important to analyze and annotate successfully Arabic corpora. This type of sentences is frequent in Arabic text and speech. Transducers can be used to realize local grammars and treat several linguistic phenomena. In this paper, we propose a parsing approach for Arabic nominal sentences using transducers. To do this, we first study the typology of the Arabic nominal sentence indicating its different forms. Then, we develop a set of lexical and syntactic rules dealing with this type of sentences and respecting the specificity of the Arabic language. After that, we present our parsing approach based on the transducers and on certain principles. In fact, this approach allows the annotation of Arabic nominal sentences but also the Arabic verbal ones. Finally, we present an idea about the implementation and experimentation of our approach in NooJ platform. The obtained results are satisfactory.

Keywords. Arabic nominal sentence, disambiguation process, topic, transducer, NooJ platform.

1 Introduction

The automatic annotation of corpora has an important place in the field of Natural Language Processing (NLP) but it has not yet taken its chance for the Arabic language. Due to its morphological, syntactic, phonetic and phonological characteristics, Arabic is considered as one of the difficult languages to analyze in NLP. So, the study of its linguistic phenomena is a necessary task. In what follows, we are interested in the study and analysis of the Arabic nominal sentences that have a close relationship with the verbal ones.

For a successful processing, we need a rigorous study of the Arabic language and especially the nominal sentence to facilitate the identification of disambiguation's rules. In fact, for the rule-based approach, there are many theoretical frameworks allowing the formalization, such as formal grammars and finite state automata [16]. Thanks to finite automata and particularly transducers, several linguistic phenomena (i.e., morphological analysis, the recognition of named entities) are treated appropriately. Transduction on text automata is useful: it can remove paths representing morph-syntactic ambiguities. In addition, lexical disambiguation can be a way to POS-tagging automatically Arabic corpora. Linguistic platforms like NooJ can facilitate the implementation and the experimentation of transducers allowing the parsing and annotation of corpora.

Indeed, the automatic annotation of Arabic nominal sentence is not an easy task. In fact, the Arabic language, that is an agglutinated one, is strongly inflected and derivational language and rich in complex structures. Moreover, the disambiguation process becomes more difficult with the absence of vocalization that is a frequent phenomenon in Arabic. To identify all structures of the Arabic nominal sentences, we make a study through a corpus and we have to study verbal sentences, because a close relationship between these two forms exists. From this study, we want to achieve a system of syntactic rules allowing the annotation of Arabic nominal sentences but also verbal ones.

A similar effort should be done for the formalization step. In fact, the formalization of elaborated rules requires much effort to guarantee several qualities (i.e., optimization, recursion, without rule's explosion and ambiguities). However, the existing linguistic platforms adapted for the Arabic language (i.e., NooJ) suffer generally from the absence of linguistic resources like dictionaries, grammars, and corpora. Always, they do not provide well-developed tools (i.e., segmentation tool).

In this context, our objective is to propose a parser for Arabic nominal sentences. This parser is implemented in the NooJ platform and it is based on a transducer cascade. The implemented parser allows also the automatic annotation of sentences. The annotation consists in associating to each sentence its syntactical representation.

In this paper, we begin by a state of the art presenting the various previously existing works that are interested in the parsing and annotation of the Arabic language. Next, we perform a study about the specificities of Arabic nominal sentences. Then, we establish transducers representing lexical and syntactic rules. Also, we implement and test all these rules in the NooJ platform. Finally, we end the present paper with a conclusion and some perspectives.

2 Previous Work

Many works aim to analyze Arabic corpora with different approaches: rule-based, statistical or hybrid approach. In [14], the authors have proposed a method for Arabic lexical disambiguation based on the hybrid approach. In [2], the author has developed a morphological syntactic analyzer for the Arabic language within Lexical Functional Grammar formalism. The developed parser based on a cascade of finite-state transducers and a set of syntactic rules specified in Xerox Linguistics Environment. Also, in [1], the authors have proposed a rule-based approach for tagging non-vocalized Arabic words. In [6], authors have designed an automatic tagging system by adding the part-of-speech tag in the Arabic text. In addition, in [8], the authors have presented an Arabic parser for Arabic nominal

sentences. In this work, the HPSG formalism is used.

In addition, there are many other statistical and hybrid works. In [4], the proposed method of parsing dealt with the 'alif-nûn' sequence in a given sentence. This method based on the context-sensitive linguistic analysis to select the correct sense of the word in a given sentence without doing a deep morpho-syntactic analysis.

Besides, in the last decades, many researchers have worked on systems, which aim to disambiguate Modern Standard Arabic. Among those systems, we mention MADA and TOKAN systems [11]. They are two complementary systems for Arabic morphological analysis and disambiguation process. Their applications include high-accuracy part-of-speech tagging, discretization, lemmatization, disambiguation, stemming and glossing.

In [5], the system AMIRA developed at Stanford University includes a tokenizer, a part of speech tagger (POS) and a Base Phrase Chunker (BPC). The model used by AMIRA is a supervised learning machine with no explicit dependence on knowledge of deep morphology. Concerning the finite state tools, we find the Xerox parser [3], which based on finite state technology, tools (e.g. xfst, twolc, lexc,) for NLP. These tools have used to develop the morphological analysis, tokenization, and shallow parsing of a wide variety of natural languages.

Moreover, there are several parsing works performed with the NooJ platform. In [15], the authors proposed a method to identify all possible syntactic representations of the Arabic relative sentences. The authors explain the different forms of relative clauses and the interaction of relatives with other linguistic phenomena such as ellipsis and coordination. We can cite also the work described in [7] to analyze the Arabic broken plural. This work is based on a set of morphological grammars used for the detection of the broken plural in Arabic texts. In fact, those transducers are the basis for a tool generated by the linguistic platform. Arabic broken plural analysis can facilitate the disambiguation process because we can distinguish between the different types of nouns.

3 Arabic Lexical Ambiguity

The Arabic language is written and read from right to left. The alphabet has 28 consonants adopting different spellings according to their position (at the beginning, middle or end of a lexical unit). Arabic token is written with consonants and vowels. The vowels are added above or below the letters. The presence of vowels allows us to understand text and disambiguate different words. In Arabic, the word should respect a well-defined type hierarchy. Indeed, a word can be either a verb or a name or a particle. Each type itself is detailed in several subtypes. Thus, any specific linguistic information to the Arabic language should be represented through this hierarchy.

Before beginning our study of lexical ambiguity, we give an overview about some specificities of Arabic language. Indeed, the Arabic sentence is characterized by a great variability in the order of its words. In general, in Arabic, we put at the beginning of the sentence the word (noun or verb) on which we want to attract the attention and at the end the richest term to keep the meaning of the sentence. This variability in the order of words causes artificial syntactic ambiguities. So in the grammar, we should give all possible combinations of inversion rules for the word order in the sentence. Note that the Arabic sentence can be either verbal or nominal.

Several aspects cause Arabic lexical ambiguity. In what follows, we focus specially on five Arabic ambiguity's causes.

Unvocalization. It can cause lexical ambiguities because a word in Arabic language can be read differently in a sentence, depending on its context. For example, the word *kaataba* can refer to the noun of the writer, or the verb to write in English.

Emphasis sign (Shadda ◌ّ). In Arabic, the emphasis sign Shadda is equivalent to write the same letter twice. The insertion of *Shadda* can change the meaning of the word. For example, the word *darasun* means *lessons* (noun) while *darrasa* means *he taught (verb)*.

Hamza sign. The presence of Hamza sign (*hamzah*) reduces ambiguity. If we add the *Hamza* to a word then the number of ambiguities decreases. As an example, the word *Faas* can be a *city*.

Agglutination. In Arabic, particles, prepositions, pronouns, can be attached to adjectives, nouns, verbs and particles. This characteristic can generate many types of lexical ambiguity. For example, the character faa' in the word *fa-slun* (*season*) is a part of the root while in the word fasala (*then he prayed*) is a prefix.

Compound words. Lexical ambiguity comes sometimes from compound words. For example, the compound noun "الحاسوب المحمول" *hassub mahmul* can be interpreted as a *laptop* o a *portable pc*.

4 Typology of Nominal Sentence

As we have mentioned, the Arabic language has two types of sentences: the nominal sentence and the verbal one. In the following section, we will present the typology of the nominal sentence. The nominal sentence is any sentence beginning with a noun and can contain a verbal sentence as a component. In addition, each nominal sentence is composed of a topic (*Mubtada* ') and an attribute (*Khabar*) and the attribute is compatible with the topic in gender and number. From this definition, we can identify several types of the Arabic nominal sentence.

4.1 Structure of Nominal Sentence

The topic and the attribute can be presented in many forms. In what follows, we detail these forms.
a. Forms of topic In our study, we concluded that the topic can have many forms. It can be a single word, a phrase or a sentence.

The case of a single noun*:* In this case, the topic can be a proper noun (name of person, geographical name, …) or a common noun. Also, it can be a personal pronoun, a demonstrative pronoun or an interrogative pronoun. Examples from (1) to (4) illustrate this case:

(1)   مريم جميلة

Mariam [is] beautiful

(2)   الطاولة مستديرة

The table [is] round

(3)   أنت جميلة

You are beautiful

(4)   هذا صديقي

This is my friend

The case of a nominal phrase: In this case, the topic can be a phrase of annexation, adjectival phrase, a relative clause or a phrase of conjunction. In addition, each one of those phrases can be recursive or contains one of the other. To illustrate this case we present the following examples:

(5)   باب الحديقة جميل

The door of the garden is beautiful

(6)   باب حديقة المنزل جميل

The door of the garden's home is beautiful

The example (5) presents a phrase of annexation which is composed of an indefinite noun (باب) and a definite noun (الحديقة), but the example (6) presents a recursive phrase which contains another phrase of annexation (حديقة المنزل) and (حديقته).

b. Forms of attribute. The attribute is manifested in several forms. It can be a unique word, a phrase or a verbal sentence.

The case of a unique word: In this case, the attribute can be a noun, a personal pronoun, an intransitive verb, or an adjective. We illustrate this case by examples (7) and (8).

(7)   العلم نور

Knowledge is the light

(8)   الولد يضحك

The boy laughs

The case of a phrase: Generally, the attribute is in the form of a phrase. It can be a nominal phrase (example (9)), a prepositional phrase (example (10)) or a relative phrase (example (11)).

(9)   الهدهد طائر جميل

Hoopoe is a beautiful bird

(10)   الأولاد في المدرسة

Boys are at school

(11)   محمد الذي نجح في الامتحان لمثابرته

Mohammed who passed the exam, due to his perseverance

The case of a sentence: the attribute can be a verbal sentence or a nominal sentence. To illustrate this case, we present the following examples:

(12)   المديرة تمنح التلاميذ المميزين الجوائز

The director gives presentes to distinguished students

(13)   الله هو النور الأعظم

Allah is the greatest light

In the example of (12), the attribute is a verbal sentence. On the other hand, the attribute of example (13) is a nominal sentence.

4.2 Other Types of Nominal Sentence

In Arabic, the nominal sentence can be introduced by particles such as the particle Inna or defective verbs such as the verb Kaana. The insertion of defective verbs or particles in a nominal sentence can change the joint of the topic and the attribute. In fact, the particle Inna accepts a subject and a predicate through dependencies called Ism inna (اسم إن) and khabar inna (خبر إن). The subject ism inna is always in the accusative case manṣūb (منصوب) and the predicate khabar inna is always in the nominative case marfū. The example of sentence (14) uses the particle Inna the topic becomes accusative but the attribute stays nominative. The same sentence of (15) without the particle Inna keeps its characteristics.

(14)   إنَّ الإمتحانَ صعبٌ

The exam is difficult

(15)   الإمتحانُ صعبٌ

The exam is difficult

5 Formalization of Lexical Rules

We carried out a linguistic study, which allows us to identify lexical rules resolving several forms of ambiguities. The identified rules were classified

Table 1. Transitivity summary table

Verb valency	Followed structures
Intransitive	NP (adverb) (PP)*
Transitive	NP NP (adverb) (PP)*
Double transitive	NP NP NP (adverb) (PP)*
Triple transitive	NP NP NP NP (adverb) (PP)*
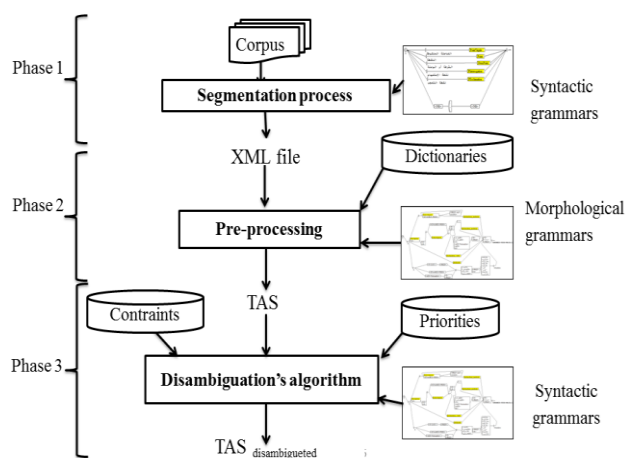
Fig. 1. Proposed method

through the mechanism of subcategorization for verbs, nouns and particles [9].

5.1 Rules for Particle

Particles can be subdivided into three categories: particles acting on nouns, particles acting on verbs and particles acting on both nouns and verbs.

Particles acting on nouns:

There are Arabic particles, which must be followed by a noun like prepositions, and particles of restriction { مِنْ، إلى، عنْ، على، في، ب، لِ، ك، حتَّى، رُبَّ، واو القسم، ت، حاشا، خلا،عدا }.

Particles acting on verbs:

Particles can also be followed by a verb, like subjunctive particles, apocope particles, prohibition particles. As an example, if we find a subjunctive particle like { أن/ لن/ كي/ حتى/ لام التعليل/

إذن/ فاء السببية/ وأو المعية/ لام الجحود }, then, it should be followed by a verb.

Particles acting on both nouns and verbs:

A noun or a verb like particles of coordination or particles of explanation can follow some particles. To solve this ambiguity, we studied the context of the sentence.

5.2 Rules for Verbs

We can apply the principle of sub-categorization to resolve the ambiguity linked to verbs. We are based essentially on the transitivity feature of verbs. In Arabic, a verb can be intransitive, transitive, di-transitive and tri-transitive. Either transitive or intransitive verbs can be transformed to transitive verbs with prepositions.

The mechanism of transitivity that is illustrated by the above sentences is summarized in the

Table 2. Table summarizing morphological grammars

Morphological grammars	Numbers
Verb inflected form patterns	113
Inflected relative pronoun patterns	8
Broken plural patterns	10
Agglutination's grammars	3

following table. Note that these examples respect the VSO order.

5.3 Rules for Nouns

We can also apply the principle of sub-categorization to resolve the ambiguity linked to nouns. We are based essentially on successors feature of nouns. In Arabic, a noun can be defined with 'ال' 'alifLam' or be indefinite. Each one of these types has their followers. Defined noun can be followed by a noun phrase (NP), a defined adjectival phrase (AP), a prepositional phrase (PP), a relative phrase (RP) or an empty set ($\varnothing$). Besides, the non-defined noun can have the same followers but the AP should be non-defined. Note that, the nominal group will inherit the nominative, accusative or genitive criteria.

To implement our rules, we use the linguistic platform NooJ that is a linguistic environment to build and manage electronic dictionaries and grammars with wide coverage and to formalize various language levels: spelling, inflectional morphology and derivational, lexicon of simple words, compound words and idioms, local syntax and disambiguation, structural and transformational syntax, semantics and ontologies. In addition, formalized descriptions can be used to process and analyze texts and large corpus.

6 Proposed Method

Our proposed approach of disambiguation consists of three main phases: the segmentation, the preprocessing and the disambiguation.

The segmentation phase [10] consists of the identification of sentences based on punctuation signs. An XML tag delimits each identified sentence. As an output of this phase, we obtain an

XML document for the corpus, and it will be the input for the pre-processing phase. The second phase consists of the agglutination's resolution using morphological grammars. As an output of this phase, we obtain a Text Annotation Structure (TAS) containing all possible annotations for corpus's sentences. The obtained TAS is the input of the third phase. The last phase aims to identify the appropriate lexical category of each word in the sentence and to construct different sentence phrases. This identification is based on several syntactic grammars specified with NooJ transducers. Transducer's applications respect a certain priority from the most evident and intuitive transducer until arriving at the least one (Figure.1). The output of the disambiguation phase will be a disambiguated TAS containing right paths and right annotations. Note that, we used a high granularity's level for lexical categories. This distinction between nominative, accusative and genitive modes for nouns can resolve the absence of vocalization.

7 Implementation

The extracted rules have been implemented in the NooJ platform [13]. The process of disambiguation of text automaton is based on the set of the developed NooJ transducers. This set contains 70 syntactic grammars representing lexical rules. In this part, we will explain different stages in our approach.

7.1 Segmentation Phase

The implementation of the segmentation phase is based on a set of developed transducers in the NooJ linguistic platform. This set contains nine graphs representing contextual rules. The main transducer adds an XML tags <S> to delimit the frontiers of a sentence.

7.2 Preprocessing Phase

The implementation of the preprocessing phase is based on a set of morphological grammars and dictionaries [12] existing in the NooJ linguistic platform (Table 2). This implementation resolves all forms of agglutination. The outputs contain all possible lexical categories of words in sentences.
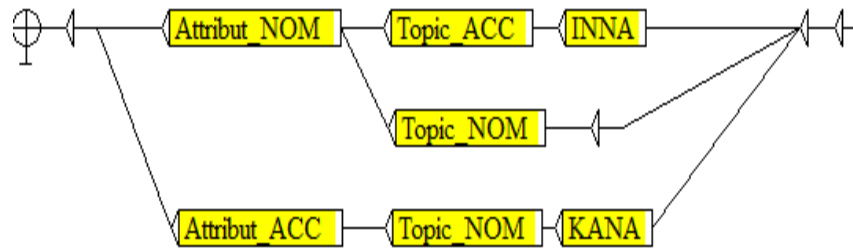
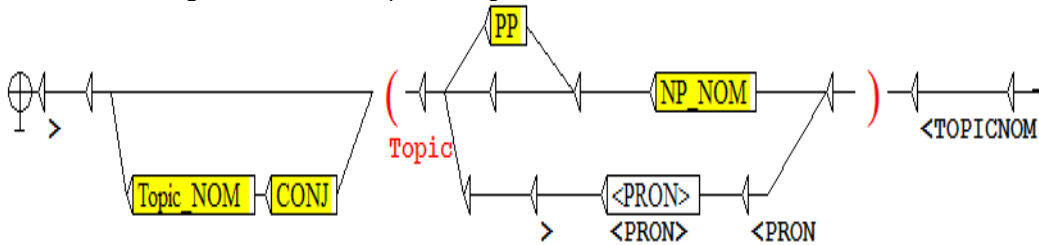**Fig. 2.** Transducer representing a lexical rule for a nominal sentence

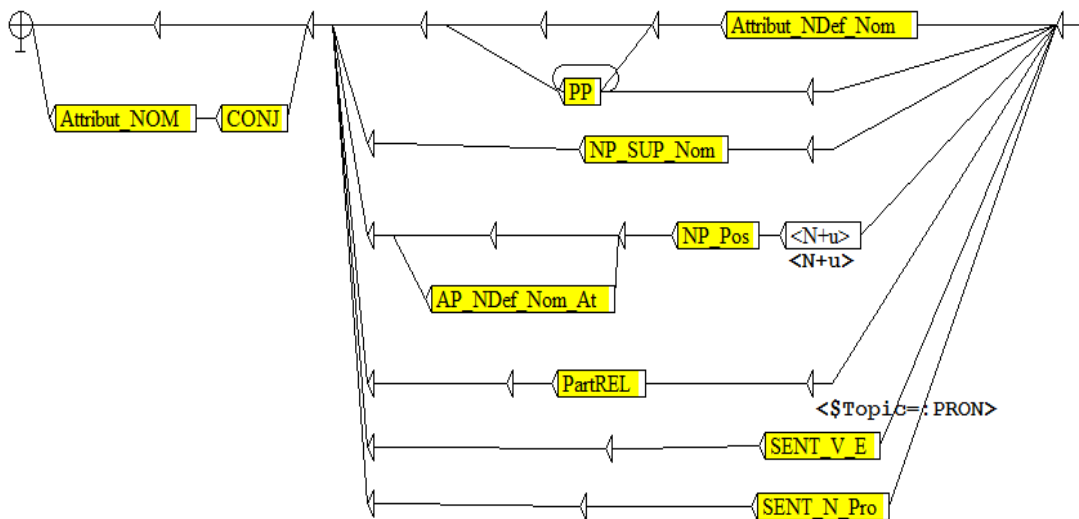**Fig. 3.** Transducer representing a lexical rule for a nominative topic

Fig. 4. Transducer representing a lexical rule for a nominative attribute

Fig. 5. Annotation of the example (16)

7.3 Analysis Phase with Disambiguation

Figure 2 illustrates the NooJ implementation of syntactic rules for nominal sentences.

In fact, Figure 2 shows different forms of topics and attributes. A nominal sentence can be formed by a nominative topic followed by a nominative attribute. Also, we can find the modal verb "KANA" followed by a nominative topic and an accusative attribute. In addition, we find the modal verb "INNA" followed by an accusative topic and a nominal attribute. In the case of a simple nominal phrase, the topic and the attribute should have the same joint they respect the nominative form.

As we have mentioned, the topic can be a unique word, a unique noun phrase or recursive one. Note that the subgraph PP represents the different forms of prepositional phrases and the subgraph NP_NOM represents the noun phrase. For a nominative attribute, the appropriate transducer is given in Figure 4.

8 Experimentation and Evaluation

To explain more our approach, we will present an example of a nominal sentence annotated with NooJ transducers.

(16) كأنّ أمرا خطيرا قد وقع صباحا
It seems that a dangerous event has occurred this morning

This example is a nominative sentence, starting with the accusative particle 'كأنّ', which is one sister of 'Inna'. This particle should be followed by an accusative topic and a nominative attribute. The topic is an adjectival phrase and the attribute is a verbal sentence where the subject was omitted. After applying our transducers for the nominal sentence, we obtained the following results illustrated in Figure 5. Note that we have used our own tag set to annotate texts.

Now we will discuss the results of our method to automatically annotate Arabic nominal sentence. To experiment our implemented prototype, we used an Arabic test corpus that contains 200 meaningful nominal sentences mainly from stories. Also, we used dictionaries containing 24,732 nouns, 10,375 verbs and 1,234 particles [12]. Besides, we used in our experimentation a list of morphological grammars containing 113 inflected verb patterns, 10 broken plural patterns and one agglutination grammar with 40 subgraphs.

Therefore, we used 70 graphs representing lexical rules, and a set of 10 constraints describing the execution of rule's application. The obtained result is illustrated in Table 3.

Table 3 shows that the partial disambiguation is due to the lack of semantic rules or to the insufficiency of granularity. Also, sometimes some rules were not correctly recognized. The erroneous disambiguation is linked also to the lack of some information in our dictionaries, which leads to the wrong detection of left or right contexts. Since, we need to elaborate other rules for different granularity levels. To illustrate the granularity

Table 3. Table summarizing the obtained result

Corpus	Number	Percentage
Sentences	200	100%
Totally disambiguated	159	80 %
Partial disambiguated	26	13 %
Failed disambiguation	15	7 %

Table 4. Table summarizing the granularity's results

Granularity's level	Number of paths	Execution's time (sec)
Level 0	1275	9.8
Level 1	956	7.6
Level 2	652	5.6
Level 3	480	4.9

Table 5. Table summarizing the precision and recall measures

Corpus	Precision	Recall	F-measure
200	0,8	0,9	0,72

effect, we will apply our syntactic grammars with different levels of granularity on the following sentence:

شاهد الولد الذي نجح

He greets the boy who succeeded

Table 4 shows the improvement result through the change of granularity level. To evaluate our prototype, we calculate also the precision, the recall and the F-measure as illustrated in Table 5.

The obtained values of these measures are ambitious and can be improved by adding other rules and heuristics.

9 Conclusion

In this paper, we have established a parsing method for Arabic nominal sentences based on a set of lexical and syntactic rules, and a high level of granularity. This method is implemented in the NooJ platform. The elaborated parser integrates also a disambiguation process and annotates automatically Arabic corpora. Therefore, we conducted a study on the different types of Arabic lexical ambiguities and on different forms of Arabic nominal sentences.

This study allowed us to establish a set of rules for parsing Arabic nominal sentences. The established rules are specified with NooJ transducers. This disambiguation phase integrated in the parser help us to reduce the parsing complexity. Thus, an experiment is performed on a set of nominal sentence mainly from stories. The obtained results are satisfactory proved by the calculated measures.

As perspectives, we want to enrich our linguistic resources by improving our dictionaries and transducers. In addition, we want to extend this methodology to other phenomena like coordination and ellipsis.

References

1. **Al-Taani, A. & Al-Rub, S. Ab. (2009).** A Rule-Based Approach for Tagging Non-Vocalized Arabic Words, International Arab. *Journal of Information Technology (IAJIT),* Vol. 6, No. 3, pp. 320–328.

2. **Attia, M. (2008).** *Handling Arabic Morphological and Syntactic Ambiguity within the LFG Framework with a View to Machine Translation.* Vol. 279.

3. **Beesley, K. (2001).** Finite-State Morphological Analysis and Generation of Arabic at Xerox. *Coling'96, Proceedings of the 16th conference on Computational linguistics,* pp. 89–94.

4. **Dichy, J. & Alrahabi, M. (2009).** Levée d'ambiguité par la methode d'exploration contextuelle: la sequence 'alif-nûn' en arabic. *Proceeding of the 2nd International Conference SIIE*, pp. 573–585.

5. **Diab, M. (2009)** Second Generation Tools (AMIRA 2.0): Fast and Robust Tokenization, POS tagging, and Base Phrase Chunking. *MEDAR 2nd International Conference on Arabic Language Resources and Tools*, pp. 285–288.

6. **Diab, M., Hacioglu, K., & Jurafsky, D. (2004).** Automatic Tagging of Arabic Text: From Raw Text to Base Phrase Chunks. *Proceeding HLT-NAACL*, pp.149–152.

7. **Ellouze, S., Haddar, K., & Abdelhamid, A. (2009).** Etude et analyse du pluriel brisé arabe avec la plateforme NooJ. *NooJ Conference and Workshop,* Tozeur, Tunisia, pp. 6–17.

8. **Haddar, K. Abdelkader, A., & Ben, H.A. (2006).** Etude et analyse de la phrase nominale arabe en HPSG. *Traitement Automatique des Langue Naturelles,* Louvain, UCL Presses de Louvain, pp. 379–388.

9. **Hammouda, G.N. & Haddar, K. (2015).** *Toward the Resolution of Arabic Lexical Ambiguities with Transduction on Text Automaton*, pp. 37–42.

10. **Hammouda, G.N. & Haddar, K. (2016).** Integration of a segmentation tool for Arabic corpora in NooJ platform to build an automatic annotation tool. *Communications in Computer and Information Science*, Vol. 667, pp. 89–100.

11. **Habash, N., Rambow, O., & Roth, R. (2009).** MADA+TOKAN: A toolkit for Arabic Tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization. *Proceedings of the Second International Conference on Arabic Language Resources and Tools*, Cairo, Egypt, pp. 102-9.

12. **Mesfar, S. (2008).** *Thesis: Analyse morpho-syntaxique automatique et reconnaissance des entités nommées en arabe strandard.* University of Franche Comté, 235.

13. **Silberztein, M. (2010)**. Disambiguation Tools for NooJ. *International NooJ Conference, Cambridge Scholars Publishing: Newcastle*, pp. 158–171.

14. **Shaalan, K., Othman, E., & Rafea, A. (2004).** Towards Resolving Ambiguity in Understanding

Arabic Sentence. *Proceedings of the International Conference on Arabic Language Resources and Tools, NEMLAR,* Cairo Egypt, pp. 118–122.

15. **Zalila, I. & Haddar, K. (2011).** Construction of an HPSG Grammar for the Arabic Relative Sentences. *Natural Language Processing, RANLP*, pp. 12–14.

16. **Gross, M. (1997).** The construction of local grammar. *Finite-State Language Processing,* England: MIT Press, pp. 329–354.