# Parsing Modern Standard Arabic using Treebank Resources

**Mostafa Al-Emran**
Al Buraimi University College
Al Buraimi, Oman
The British University in Dubai
Dubai, UAE
malemran@buc.edu.om

**Sarween Zaza**
The British University in Dubai
Dubai, UAE
120155@student.buid.ac.ae

**Khaled Shaalan**
The British University in Dubai
Dubai, UAE
khaled.shaalan@buid.ac.ae

***Abstract*—A Treebank is a linguistic resource that is composed of a large collection of manually annotated and verified syntactically analyzed sentences. Statistical Natural Language Processing (NLP) approaches have been successful in using these annotations for developing basic NLP tasks such as tokenization, diacritization, part-of-speech tagging, parsing, among others. In this paper, we address the problem of exploiting Treebank resources for statistical parsing of Modern Standard Arabic (MSA) sentences. Statistical parsing is significant for NLP tasks that use parsed text as an input such as Information Retrieval, and Machine Translation. We conducted an experiment on Pen Arabic Treebank (PATB) and the parsing performance obtained in terms of Precision, Recall, and F-measure was 82.4%, 86.6%, *84.4%*, respectively.**

***Keywords— Statistical Parsing; Treebank; Arabic.***

## I. INTRODUCTION

Arabic is a Semitic language that is characterized by its rich morphology and complex syntax [1], [5], [6], [17], [18]. Parsing Arabic sentences is one of the challenging NLP tasks due to the peculiarities and distinct features of Arabic [2]. The difficulties in parsing Arabic sentences comes from different aspects of the language such as syntax complexity, sentence length, diacritics and punctuation omissions, existence of the elliptic personal pronoun and the free word order [3]. Failing to successfully address these issues would result in either inaccurate or ambiguous parsing [19].

Linguistic Data Consortium (LDC) has developed Treebanks as high quality linguistic resources in several languages, including Arabic [4]. Traditional Arabic grammar theories of MSA, in addition to the modern grammatical theories, were the basis of the annotation that is applied to the PATB. Our statistical parsing approach involves using the PATB's annotation patterns for generating a statistical model that can predict the parsing of an input Arabic sentence.

In this paper, we address exploiting Treebank resources for parsing Arabic sentence. PATB's parse trees are used to train the statistical parser. It is able to get an Arabic sentence as input and generates a successful parse tree as output. A convenient Graphical User Interfacing is also developed. Our aim is to improve the performance of NLP tasks that uses parsed text as an input, such as Information Retrieval, question answering, and Machine Translation. The system has been evaluated using a gold standard dataset from PATB apart from the training dataset. The experimental results obtained were a Precision of 82.4%, a Recall of 86.6%, and an F-measure of 84.4%.

## II. BACKGROUND

### A. Penn Arabic Treebank

PATB is offered by LDC since 2001 [4]. PATB's project has been released in three full versions that is annotated by linguistics features: 1) Part 1 v 2.0, with about 166K words written MSA newswire, was built in two stages: morphological annotation and syntactic annotations, which is from the Agence France Presse corpus, 2) Part 2 v 2.0, with nearly 144K words from Al-Hayat newspaper, and 3) Part 3 v 1.0, contains nearly 350K words of newswire text from An-Nahar newspaper, were morphologically annotated. The second and third versions were mainly focused on improving the parsing accuracy [7]. The project was built to support the morphological analyzers data-driven and syntactic parsers, with 23,611 parsed annotated sentences from MSA. Lately, the PATB has become a linguistic resource for many parsers, including the Bikel's statistical parser [8].

### B. Statistical Parsing

Parsers are primarily developed using one of the following approaches: Rule-based (linguistic-based) approach [5] and Machine Learning-based approach [8]. Rule-based parsers rely on handcrafted grammatical rules written by linguists. The main advantage of the rule-based parsers is that they are based on a core of solid linguistic knowledge [5]. However, any maintenance or updates required for these systems is labor-intensive and time consuming. On the other hand, machine learning-based parsers utilize learning algorithms that require large tagged datasets (Corpora or Treebank) for training and testing. Machine learning algorithms involve a selected set of features extracted from datasets annotated with linguistic

978-1-4799-8966-9/15/$31.00 ©2015 IEEE

features in order to generate statistical models for parse prediction [19]. An advantage of the machine learning-based parsers is that they are adaptable and updatable with minimal time and effort as long as sufficiently large annotated datasets are available [18].

### C. Limitations of Statistical Parsers

Historically, successful statistical parsers have been developed for parsing English sentences. However, in [10], the following limitations were identified: 1) A fixed reasonable probabilistic structure having the capacity to be changed through recoding few considerable portion of the program, 2) Hard coded for particular characteristic framework to English, 3) Hard coded for particular characteristic framework to the Penn Treebank, and 4) Designed exclusively for uniprocessor environment setting. The solution was the design of Bikel parser, which has addressed all the aforementioned limitations. Bikel has developed an extensible, parallel statistical parsing that implements several types of generative, Collins' statistical parsing models. It has a "plug and play" probability structure. It can also be ported to new domains and languages, including Arabic.

## III. Related Work

Parsing Arabic sentences is a challenging NLP task. The main reason that motivates NLP researchers to develop various parsers augmented with various methodologies is to reach as much as highly possible accurate parses in the way that an Arabic linguist would.

In [11], a statistical parser is described which is built on the dependencies' probabilities between words in the parse trees. This parser takes a tagged sentence as an input and generates a phrase-structure tree as an output. The parser has been trained on 40,000 sentences and tested using 2416 sentences from some parts in the Penn Treebank.

In [12], an integration of three parsers based on two main methodologies, *parser hybridization* and *parser switching,* was presented. The aim is to attain high parsing accuracy by minimizing the individual parsers' errors and distributing these errors independently. The three parsers were trained and tested on several portions of the Penn Treebank.

An efficient chart-parser was developed [3], which is capable of analyzing MSA sentences. The developed parser has the ability to satisfy syntactic constraints in order to reduce the parsing ambiguity along with the lexical semantic features that are used to disambiguate the sentence structure.

In [9], the authors illustrated the parsing of Arabic using two methodologies: automatic Lexical Functional Grammar (LFG) parsers that accept f-structure annotation and Treebank-based parsers. The Arabic Annotation Algorithm ($A^3$) has utilized the rich functional annotations of the PATB in order to associate f-structure with parse trees. A modification has been done on the Bikel's parser in order to learn the PATB functional tags and to combine the phrasal categories along with the functional tags in the training dataset. The output of the Bikel's parser has been automatically enriched in the form of LFG f-structure via ($A^3$). The results achieved 77% as a dependency f-measure score.

A parsing approach was developed in [2], which parses MSA sentences in general and Quranic chapters (*Suras*) in particular, in order to generate parse trees through the use of Natural Language Toolkit (NLTK) library of Python. The approach has been accomplished through constructing a top-down parser, and acquiring a lexicon and a context-free grammar.

## IV. Methodology

### A. The Development Environment

The Microsoft Visual studio 2008 is used to implement the user interfaces and connecting them with the *training* and *parsing* modules.

### B. Data Analysis

The dataset has been divided into two parts (training and testing datasets) and used to characterize grammar features by the Arabic Treebank annotations.

#### *The Training Dataset*

The parser in this paper has been trained on about 22000 Arabic sentences from PATB.

For example, the Arabic sentence:

**"ويبلغ إعداد المشردين في كاونتى لوس انجلوس نحو 84 ألف شخص بينهم تسع آلاف طفل"**

has the following analysis in the PATB which is used to train the parser:

```
(S (CONJ wa-)(VP (IV3MS+IV+IVSUFF_MOOD:I -ya+bolug+u ويـبـلـغ)
(NP-SBJ (NOUN+CASE_DEF_NOM Eadad+u إعداد)
(NP (DET+ADJ+NSUFF_MASC_PL_GEN Al+mu$ar~ad+iyna المشردين)))
(PP-LOC (PREP fiy في)
(NP (NOUN+NSUFF_FEM_SG+CASE_DEF_GEN kuwnotiy~+ap+I كـاونـتي)(NP (NOUN_PROP luws لـوس)(NOUN_PROP >anojiliys أنجـلـوس))))
(NP-OBJ (NP (QP (PREP naHowa نحو)
(NUM 84 84)(NOUN+CASE_DEF_ACC >alof+a ألـف))
(NOUN+CASE_INDEF_GEN $axoS+K شخص))
(PRN (S (PP-PRD (PREP bayona- بـين)(NP (PRON_3MP -hum هم)))
(NP-SBJ (QP (NUM+NSUFF_FEM_SG+CASE_DEF_NOM tisoE+ap+u تـسع)
(ADJ+CASE_DEF_GEN |lAf+I آلاف))(NOUN+CASE_INDEF_GEN Tifol+K طفـل)))))
```

#### *The Testing Dataset*

The system has been evaluated using a gold standard data set from the PATB apart from the training dataset. The test set

consists of 2000 transliterated sentences and its corresponding parse trees.

The outcome of each phrase has been inquired systematically to evaluate the workflow of our system using the test set.

### C. An Example

***The Translated Input***

The input Arabic surface sentence:

**"بلغت أستراليا حاملة اللقب الدورة النهائية لكأس ديفيز لكرة المضرب بتقدمها على البرازيل 3 صفر في اليوم الثاني من منافسات الدوري نصف النهائي التي تقام في بريزبان استراليا"**

is preprocessed into its transliterated format to meet the input format of the parser:

```
balag+at usoturAliyA HAmil+ap+u Al+laqab+i Al+dawor+a Al+nihA}iy~+a li -musAbaq+ap+i ka>os+i diyfiys li -kur+ap+i Al+miDorab+i bi -taqad~um+i-hA EalaY Al+barAziyl+i 3 Sifor+N fiy Al+yawom+i Al+vAniy min munAfas+At+i Al+dawor+i niSof+i Al+nihA}iy~+i Al~atiy tu+qAm+u fiy briyzobAn usoturAliyA
```

***The System Output***

The following is the output parsed sentence:

```
(S (VP (VBD balag+atusoturAliyAHAmil+ap+uAl+laqab+iAl+dawor+aAl+nihA}iy~+a) (PP (IN li) (NP (NNP -musAbaq+ap+ika>os+idiyfiys))) (PP (IN li) (NP (NNP -kur+ap+iAl+miDorab+i))) (SBAR (IN bi) (S (VP (VBP -taqad~um+i-hAEalaYAl+barAziyl+i) (NP (NP (CD 3) (NN Sifor+NfiyAl+yawom+iAl+vAniy)) (PP (IN min) (NP (NN munAfas+At+iAl+dawor+iniSof+iAl+nihA}iy~+iAl~atiytu+qAm+ufiybriyzobAnusoturAliyA))))))))) (PUNC .))
```

***The Gold Standard***

The extracted gold standard of the input sentence from the PATB is:

```
(S (VP (PV+PVSUFF_SUBJ:3FS balag+at)(NP-SBJ (NP (NOUN_PROP >usoturAliyA))(NP (PUNC ,)(ADJ+NSUFF_FEM_SG+CASE_DEF_NOM HAmil+ap+u)(NP (DET+NOUN+CASE_DEF_GEN Al+laqab+i)(PUNC ,))))(NP-OBJ (NP (DET+NOUN+CASE_DEF_ACC Al+dawor+a)(DET+ADJ+CASE_DEF_ACC Al+nihA}iy~+a))(PP (PREP li-)(NP (NP (NOUN+NSUFF_FEM_SG+CASE_DEF_GEN -musAbaq+ap+i)(NP (NOUN+CASE_DEF_GEN ka>os+i)(NP (NOUN_PROP diyfiys))))(PP (PREP li-)(NP (NOUN+NSUFF_FEM_SG+CASE_DEF_GEN -kur+ap+i)(NP (DET+NOUN+CASE_DEF_GEN Al+miDorab+i)))))))(PP (PREP bi-)(NP (NP (NOUN+CASE_DEF_GEN -taqad~um+i-)(POSS_PRON_3FS -hA))(PP (PREP EalaY)(NP (DET+NOUN_PROP+CASE_DEF_GEN Al+barAziyl+i)))))(ADVP (NUM 3)(PUNC -)(NUM+CASE_INDEF_NOM Sifor+N))(PP-TMP (PREP fiy)(NP (NP (DET+NOUN+CASE_DEF_GEN Al+yawom+i)(DET+ADJ Al+vAniy))(PP (PREP min)(NP (NP (NOUN+NSUFF_FEM_PL+CASE_DEF_GEN munAfas+At+i)(NP (DET+NOUN+CASE_DEF_GEN Al+dawor+i)(NP (NOUN+CASE_DEF_GEN niSof+i)(DET+ADJ+CASE_DEF_GEN Al+nihA}iy~+i))))(SBAR (WHNP-1 (REL_PRON Al~atiy))(S (VP (IV3FS+IV_PASS+IVSUFF_MOOD:I tu+qAm+u)(NP-SBJ-1 (-NONE- *T*))(NP-OBJ-1 (-NONE- *))(PP-LOC (PREP fiy)(NP (NP (NOUN_PROP briyzobAn))(PRN (PUNC -LRB-)(NP (NOUN_PROP >usoturAliyA))))))))))))(PUNC -RRB-)(PUNC .))
```

## V. System User Interface

A convenient GUI has been implemented in order to make the system more efficient and interactive. The GUI allows the user to enter an Arabic sentence and generates the parse tree that corresponds to the analysis of this sentence based on the statistical model that was produced from training phase, see Fig. 1.

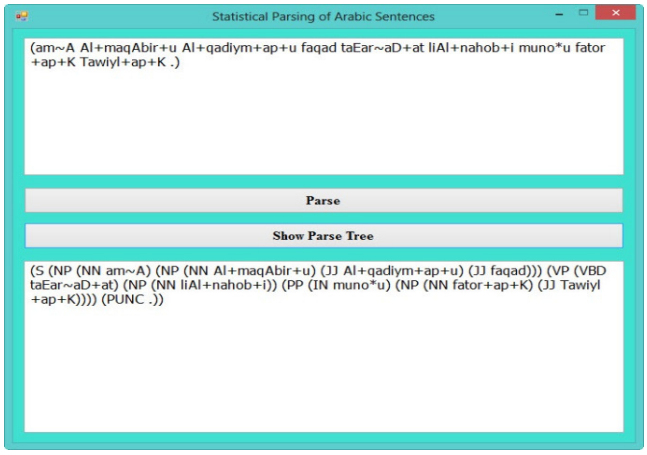


Fig. 1. Parsing interface.

## VI. Evaluation

Parsing of the PATB was first attempt by [8]. The acquisition and annotation of the data are described in [7] and [14]. Different evaluation techniques were utilized for evaluating the parsing performance. Two evaluation techniques were used. The first one compares the fundamental outcomes, e.g. the object of PP to the nearest NP/VP. Whereas, the second technique, which we followed, compares the system's output with the gold standard dataset. Due to unavailability of an automatic evaluation tool of parsed sentences, we decided to manually calculate the Precision, Recall and F-measure on a test set consists of 2000 Arabic parsed sentences. The test dataset has been randomly selected from the PATB for the testing purpose apart from the training dataset.

The standard evaluation measures [15] [16]: Precision, Recall, and F-measure are used to evaluate the performance of the system. Precision is calculated by the following equation:

$$\textbf{\textit{Precision}} = \frac{\text{Number of correct constituents in } P}{\text{Number constituents in } P}$$

Recall is calculated by the following equation:

$$\textbf{\textit{Recall}} = \frac{\text{Number of correct constituents in } P}{\text{Number constituents in } T}$$

F-measure is calculated by the taking the harmonic mean of both precision and recall as follows:

$$F - Measure = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Table 1 shows the system's performance in terms of Precision, Recall and F-measure.

TABLE I. EVALUATION RESULTS

| Precision | Recall | F-measure |
|---|---|---|
| 82.4% | 86.6% | 84.4% |

## VII. CONCLUSIONS AND FUTURE WORK

Parsing Arabic sentences is one of the crucial issues in developing NLP applications such as Machine Translation and Information Retrieval. This is the main reason that motivates us to investigate the development of a system for parsing Arabic sentences. The system has been implemented as standalone software using Visual Basic.Net and can run offline. The developed system relies on the Arabic Statistical Parser to produce a model through a training phase on the Penn Arabic Treebank, a de facto standard Arabic linguistic resource. The system provides an interactive user interface that facilitates the input of free Arabic sentences. The main design goals of the developed system include several features that make it distinct from the current available systems, such as *portability*, *local installation* and *simplicity*. The parser in has been trained on 22000 Arabic sentences from PATB. We conducted an experiment on 2000 sentences from the PATB and the parsing performance obtained was a Precision of 82.4%, a Recall of 86.6%, and an F-measure of 84.4%

For further and future research work, we plan to test the system on a larger dataset and different domains. Moreover, the system could be enhanced in a way that could accept an Arabic script from the user and generates a graphical representation showing detailed analysis of the input sentence. Nevertheless, we plan to incorporate the system in larger tasks such as information retrieval, machine translation, and question answering systems.

## REFERENCES


[1] Farghaly, A., & Shaalan, K. (2009). Arabic natural language processing: Challenges and solutions. *ACM Transactions on Asian Language Information Processing (TALIP)*, *8*(4), 14.

[2] Shatnawi, M., & Belkhouche, B. (2012). Parse Trees of Arabic Sentences Using the Natural Language Toolkit. *College of IT, UAE University, Al Ain.*

[3] Othman, E., Shaalan, K., & Rafea, A. (2003). A chart parser for analyzing modern standard Arabic sentence. In *Proceedings of the MT Summit IX Workshop on Machine Translation for Semitic Languages: Issues and Approaches* (pp. 37-44).

[4] Maamouri, M., & Cieri, C. (2002). Resources for Arabic Natural Language Processing. In *International Symposium on Processing Arabic* (Vol. 1).

[5] Shaalan, K. (2010). Rule-based approach in Arabic natural language processing. *The International Journal on Information and Communication Technologies (IJICT)*, *3*(3), p 11-19.

[6] Habash, N., Dorr, B., & Monz, C. (2006). Challenges in building an Arabic-English GHMT system with SMT components. In *Proceedings of the 11th Annual Conference of the European Association for Machine Translation (EAMT-2006)*, pp. 56 - 65.

[7] Maamouri, M., Bies, A., &Kulick, S. (2006). Diacritization: A challenge to Arabic treebank annotation and parsing. In *Proceedings of the Conference of the Machine Translation SIG of the British Computer Society*.

[8] Bikel, D. M. (2004). A Distributional Analysis of a Lexicalized Statistical Parsing Mode. In *EMNLP*, pp. 182 - 189.

[9] Attia, M., Shaalan, K., Tounsi, L., Genabith, J., Automatic Extraction and Evaluation of Arabic LFG Resources, *Proceedings of The eighth international conference on Language Resources and Evaluation (LREC'12)*, PP. 1947-1954, Istanbul, Turkey, 21-27 May 2012.

[10] Bikel, D. M. (2002, March). Design of a multi-lingual, parallel-processing statistical parsing engine. In *Proceedings of the second international conference on Human Language Technology Research* (pp. 178-182). Morgan Kaufmann Publishers Inc.

[11] Collins, M. J. (1996). A new statistical parser based on bigram lexical dependencies. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pp. 184 - 191. Association for Computational Linguistics.

[12] Henderson, J. C., & Brill, E. (1999). Exploiting diversity in natural language processing: Combining parsers. In *Proceedings of the Fourth Conference on Empirical Methods in Natural Language Processing*, pp. 187 - 194.

[13] Wikipedia, the free encyclopaedia. 2014. *Microsoft Visual Studio*[online]. [Accessed 16th February, 2014]. Available at: http://en.wikipedia.org/wiki/Microsoft_Visual_Studio

[14] Kulick, S., Gabbard, R., & Marcus, M. (2006). Parsing the Arabic treebank: Analysis and improvements. In *Proceedings of the Treebanks and Linguistic Theories Conference*, pp. 31 - 42.

[15] Manning, C. D. and Schütze, H. (2002). *Foundations of Statistical Natural Language Processing*. London: MIT Press, 2002.

[16] Jurafsky, D., & James, H. (2000). *Speech and language processing an introduction to natural language processing, computational linguistics, and speech*.

[17] Al Emran, M., & Shaalan, K. (2014). A Survey of Intelligent Language Tutoring Systems. In *Advances in Computing, Communications and Informatics (ICACCI), 2014 International Conference on* (pp. 393-399). IEEE.

[18] Shaalan, K. (2014). A Survey of Arabic Named Entity Recognition and Classification. Computational Linguistics, 40 (2), p 469-510, MIT Press, USA.

[19] Al-taher, A., Abo Bakr, H., Zidan, I. & Shaalan, K. (2014). An Arabic CCG approach for determining constituent types from Arabic Treebank, Special Issue on Arabic NLP, Journal of King Saud University - Computer and Information Sciences, Elsevier, 26(4):441 - 449.