Developing a Transition Parser for the Arabic Language

Aref abu Awad
Computer Information System, Zarqa University,
Zarqa, Jordan

Essam Hanandeh
Computer Information System, Zarqa University,
Zarqa, Jordan

***Abstract*—One of the most important Characteristics of the Arabic language is the exhaustive undertaking. Thus, analyzing Arabic sentences is difficult because of the length of sentences and the numerous structural complexities. This research aims at developing an Arabic parser and lexicon. A lexicon has been developed with the goal of analyzing and extracting the attributes of Arabic words. The parser was written by using a top–down algorithm parsing technique with recursive transition network. Then, the parser has been evaluated against real sentences and the outcomes were satisfactory.**

Keywords—Natural language processing; Arabic parser; lexicon; Transition Network

I. INTRODUCTION

Natural language processing (NLP), which is considered a field of computer science, artificial intelligence, and computational linguistics, is dealing with the interactions between computers and natural languages. Accordingly, NLP is related to the area of human–computer interaction. Many challenges in NLP involve natural language understanding, that is, enabling computers to derive meaning from human or natural language input. Other challenges involve natural language generation. The history of NLP generally started in the 1950s, although studies can be traced from periods earlier than that a decade. In 1950, Alan Turing published an article entitled "Intelligence, "which proposed what is now called the Turing test as a criterion of intelligence. Recent research has increasingly focused on unsupervised and semi-supervised learning algorithms. These algorithms are able to learn from data that have not been hand-annotated with the desired answers, or use a combination of annotated and non-annotated data. In general, this task is considerably more difficult than supervised learning and typically produces inaccurate results for a given amount of input data. However, an enormous amount of non-annotated data are available (including the entire World Wide Web content) often compensate the inferior results. Modern NLP algorithms are based on machine learning, particularly statistical machine learning. The machine learning paradigm is different from that of most prior attempts at language processing. Prior implementations of language-processing tasks typically involved the direct hand coding of large sets of rules. The machine-learning paradigm calls for using general learning algorithms, which are often grounded on statistical inference, to automatically learn such rules through the analysis of large corpora of typical real-world examples. A corpus (plural: corpora) is a set of documents (or individual sentences) that have been hand-annotated with the correct values to be learned. The goal of the NLP group is to design and develop software that will analyze, understand, and generate languages that humans can use to address a computer and addressing another person [1]. Information retrieval is one of the natural language processing applications that appears in these definitions. Information retrieval is a field which deals with the structure, analysis, organization, storage, searching, and retrieval of information [2]. Moreover, information retrieval is a selective process by which the desired information is extracted from a store of information called a database [3].

II. RELATED STUDIES

Gilbert et al. [8] developed a bottom–up parsing strategy for summarizing an English text and integrated it with the Pruner and Redundancy Eliminator (PARE) system, replacing the old link grammar parser which was previously used. Constituency trees from our parser provide all parts-of-speech linkages as input to several other code modules in the PARE system. Our parser uses rules that are written in the Chomsky normal form, which is a specialization of a general context-free grammar. Updating the PARE system leads to an increase in the efficiency of the text summarization process [8].

Shaalan et al. [10] developed an Arabic parser for modern scientific text. This parser is written in definite clause grammar and is targeted to be a component of a machine translation system. The development of the parser consisted of a two-step process. In the first step, we acquired the rules constituting the Arabic grammar that provided a precise account of what was considered a grammatical sentence. The grammar covered a text from the domain of the agricultural extension documents. The second step involved implementing the parser that assigns grammatical structure to the input sentence. An experiment on real extension document was performed, and the results observed were satisfactory.

Khufuet al. [11] recommended a method for Arabic parsing based on supervised machine learning. They used the support vector machines algorithm to select the syntactic labels of the sentence. Furthermore, we evaluated their parser following the cross validation method by using the Penn Arabic Treebank. The obtained results were substantially encouraging.

Al-Taani1 et al. [12] presented a top–down chart parser for parsing simple Arabic sentences, including nominal and verbal sentences within the specific Arabic grammar domain. We used context-free grammar (CFG) to represent the Arabic grammar. We first developed the Arabic grammar rules that

provided precise description of grammatical sentences. Thereafter, we implemented the parser that assigns grammatical structure to the input sentence. Experimental results showed the effectiveness of the proposed top–down chart parser for parsing modern standard Arabic sentences.

PARSIG METHOD

Parsing method involves revealing a structure in an input based on the external information about the elements of the input and their order. Generally speaking, external information comprises a lexicon, i.e., list of input words; and grammar to describe the structures that may be built from and implemented by the sequences of words [9]. Parsing has several definitions but most of them focus on the text structure. The common definitions of parsing are as follows. Parsing can be defined as the process of analyzing an input sequence in order to determine its grammatical structure regarding to a given formal grammar [5]. Parsing breaks a sentence down into its component parts of speech with an explanation of the form, function, and syntactical relationship of each part [6]. Parsing is also the process of converting text input into a data structure defining its syntactical structure and semantic meaning based upon a given formal grammar [8]. Parsing natural language is an attempt to discover a certain structure in a text (or textual representation) generated by a person [4]. A parser is a computational system that processes input sentences according to the productions of grammar, and builds one or more constituent structures that conformed grammatically. We consider grammar as a well-formed declarative specification, whereas a parser is a procedural interpretation of grammar.

III. LEXICON

Lexicography is the branch of applied linguistics concerned with the design and construction of lexica for practical use. Lexica can range from the paper lexica or encyclopedia designed for human use and shelf storage to the electronic lexica used in a variety of human language technology systems, such as word databases, word processors, and software for reading back (by speech synthesis in text-to-speech systems) and dictation (by automatic speech recognition systems). At a considerably generic level, a lexicon may be a generic lexicographic knowledge base from which these different types of lexica can be derived automatically [71]. Meanwhile, lexicology is the branch of descriptive linguistics concerned with linguistic theory and methodology for describing lexical information, and often focuses specifically on issues of meaning. Traditionally, lexicology has been mainly concerned with lexical collocations and idiom, lexical semantics, as well as the structure of words, meaning components and relationships between them.

IV. TRANSITION NETWORK GRAMMARS

Transition network grammar is considered as a formalism for representing grammars based on the concept of a transition network that comprises nodes and labeled arts. This formalism developed out from the transition network concept of a finite-state automaton. It is equivalent to push-down automata because the arts, comprise the network of a transition network grammar and represent transcriptions of the rules of a context-free grammar [7]. Sentences generated by the grammar are accepted by a transition network grammar through the process of traversing the network comprising of these arcs.

Figure 1 shows the network called NP in which each art is labeled with a word category. Starting at a given node, one can traverse an art if the current word in the sentence is in the category on the art. If the art is followed, then the current word is updated to the next word. A phrase is a legal NP if a path from the node NP to a pop art accounts for every word in the phrase.

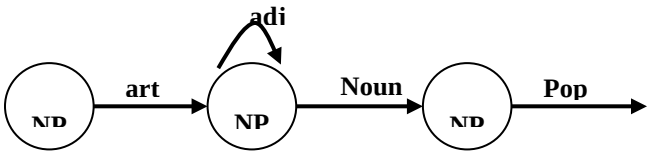

Fig. 1. Transition Network

V. SYSTEM EVALUATION

The objective of our experiment was to test whether the parser is sufficient for application to real Arabic sentences. We selected an unrestricted Arabic sentence, which is from the Arabic students' book.

VI. RESULTS

We discuss the experiment results whether the input sentence is parsable or not. Table (1) shows the results of the parser. These results are categorized into: parsable and unparsable sentences.

The parsable sentence is divided into two subcategories as follows.

1) Syntactically Correct: This subcategory led to a complete and successful parsing of the input sentence.

2) Syntactically Incorrect: This subcategory led to a complete parsing of the input sentence but the result, as can be seen, is a syntactically incorrect structure. The source of this error does not match in terms of attributes (e.g., gender, number) between words of sentence. For example, the input sentence

يذهب الطالبة إلى المدرسة

is not parsed by our parser. The subject (الطالبة) takes the female feature gender. However, the prefix (ي) of the verb (يذهب) of the sentence indicates that this feature value is for male. The syntactically correct sentence would be as follows:

تذهب الطالبة إلى المدرسة.

The unparsable sentence can be divided into three subcategories:

1) Lexical Problem: The parser does not find out the word in the lexicon.

2) Incorrect Sentence: This subcategory has failed to parse because the input sentence is incorrect:

يلعب يدرس الطالب النشيط .

3) Failure: The sentence is not identified by linguists according to Arabic grammar rules. An example is the following input sentence:

الطالب النشيط يدرس.

TABLE I. RESULTS OF THE PARSER

		Number of Sentences	Percentage
Parsable Sentence	Syntactically Correct	77	87.1 %
	Syntactically Incorrect	2	2.6 %
Unparsable Sentence	Lexical Problem	4	4.8 %
	Incorrect Sentence	2	2.4 %
	Failure	5	5.8 %
Total		93	100 %

The number of sentences used in the test was 93 and the length of each sentence was 6 words. The result shows that the number of successfully parsed sentences were 77 (87.1%) and 2 sentences were syntactically incorrect (2.6%). The number of sentences that were not parsed (i.e., has lexical problem) were 4 (4.8%). The number of sentences that were not parsed (incorrect sentence) were 2 (approximately 2.4%). The number of sentences that were not parsed (i.e., not recognized by linguists according to Arabic grammar rules) were 5 (approximately 5.8%).

VII. ANALYSIS OF RESULTS

1) Analysis of the Syntactically Incorrect Sentences

Recall that the number of syntactically incorrect sentences were 2 sentences. The parser assigned the incorrect result to the input sentence. Hence, the parser completed the sentence parsing, but the result is incorrect. This result was due to an incomplete agreement between word attributes (e.g., gender, number).

2) Analysis of the Unparsable Sentences

Recalling that the number of unparsable sentences were 11; the parser failed to identify any rule to the input sentence. These are classified into three categories as follows.

a) Lexical Problem: The parser fails to recognize any rule to the input sentence and this is because certain parts of the sentences are unavailable in the lexicon. Thus, the parser does not obtain the attributes of these parts.

b) Incorrect Sentence: The parser fails to produce a rule for the input sentence because of the incorrect syntactic form of the sentence. Hence, determining an equivalent role in the sentential form in the parser is impossible.

c) Failure: The parser fails to produce a rule for the input sentence because the syntactic form of the sentence is excluded in the grammar. Thus, failure may result when the sentence structure is correct.

VIII. CONCLUSION

Our contribution in this paper is to design, build and Evaluate system for parsing Arabic sentences and Determine if these sentences syntactically correct or not. In addition, the proposed system builds a lexicon for Arabic sentences.

The Arabic language lacks parsing systems for analyzing Arabic sentences. Parsing systems are crucial in natural language processing because they are used as a first step in most natural language processing applications. Moreover, this system can be extensively used for educational purposes.

In the natural Arabic language processing, predefined forms, exist for analyzing sentences, make parsing problematic. The Arabic sentence is complex and syntactically ambiguous because of the frequent usage of grammatical relationships, conjunctions, and other constructs.

The methodology we adopted in this study based on analyzing the Arabic language grammar conforming to gender and number, formalization of rules using CFG, representation of the rules using transition networks, constructing a lexicon of words that will be in the sentences structure, implementing the recursive transition network parser, and evaluating the system using real Arabic sentences. Finally, the current analysis was effective and provided good results

REFERENCES

[1] Preeti1, and B. Sidhu, 2013. NATURAL LANGUAGE PROCESSING. Int.J.Computer Technology & Applications,Vol 4 (5),751-758.*)*

[2] T. Strzalkowski, F. Lin, J. Wang, J. Perez-Carballo, 1999. Evaluating Natural Language Processing Techniques in Information Retrieval. TREC,Volume 7, pp 113-145.

[3] J. allan, J.Aslam, N. Belkin, 2003. Challenges in Information Retrieval and Language Modeling. ACM SIGIR Forum, 37(1):31-47.

[4] Taboada, Maite, and William C. Mann. "Applications of rhetorical structure theory." *Discourse studies* 8.4 (2006): 567-588.

[5] Kübler, Sandra, Ryan McDonald, and Joakim Nivre. 2009 Dependency parsing. *Synthesis Lectures on Human Language Technologies* 1.1 pp. 1-127..

[6] Weise, D. Neal. 2007. Method and apparatus for improved grammar checking using a stochastic parser. U.S. Patent No. 7,184,950. 27

[7] Budanitsky, Alexander, and G. Hirst. 2006. Evaluating wordnet-based measures of lexical semantic relatedness." *Computational Linguistics* vol.32.pp 13-47.

[8] Gilbert, Nathan, E. Welborn, and S. Thede. 2005 PARSING ENGLISH TEXTS IN PARE.

[9] Bird, Steven, and M. Liberman, 2001.A formal framework for linguistic annotation. *Speech communication*, pp. 23-60.

[10] Shaalan, Khaled, A. Farouk, and A. Rafea,1999.Towards an Arabic parser for modern scientific text. *Proceeding of the 2nd Conference on Language Engineering*.

[11] Elarnaoty, Mohamed, S. AbdelRahman, and A. Fahmy, 2012. A machine learning approach for opinion holder extraction in Arabic language.*arXiv preprint arXiv:1206.1011*..

[12] T. Ahmad, M. Mohammed, and A. Sana, 2012."A top-down chart parser for analyzing arabic sentences." *Int. Arab J. Inf. Technol.* 9.2,pp. 109-116.