A LINK GRAMMAR PARSER FOR ARABIC

Warren Casbeer, Jon Dehdari, and Deryle Lonsdale
BYU Department of Linguistics

1. Introduction

A parsing system is critical to many natural language processing tasks. Developing an automatic parser of Arabic is an issue that warrants further investigation, as this language has unique difficulties.

A number of problems confront current Arabic parsers. Many are computationally costly or have limited coverage. Due to these issues, Arabic parsers for use in large-scale applications are not available (Ouersighni 2001). Ambiguity is a particular issue that has not been treated to a large extent (Al Daimi 2001, Othman et al. 2003). Robustness is another area of potential improvement (Ouersighni 2001).

In this paper, we introduce a new parser for Arabic based on link grammar, a dependency-like grammar. This grammar is has been implemented as a parser that is cost efficient, widely used, and freely available. A grammar may be defined and implemented by independent users.

Link grammar has been used to provide coverage for a variety of languages. Though originally developed for English, it has been applied to other languages such as Chinese (Liu, in print), Persian (Dehdari & Lonsdale 2005), and Russian (Protasov)[1].

Ambiguity detection is also possible with this system. Multiple parses are provided by the parser as all possible structures are considered. These are presented in ranked order according to the constraints of the grammar.

[1] http://sz.ru/parser/

The parser also displays robustness since it can guess the categories of words not present in the lexicon based on its knowledge of the syntactic environment. It may also skip unknown structures that are encountered.

This paper is organized as follows. First, preprocessing is described. This includes input formats, morphological decomposition, and necessary lexicons. Afterwards, link grammar is introduced and grammar development is described. To exemplify coverage in Arabic, sample parses are shown. Ambiguity resolution using the parser is then demonstrated, and an evaluation of the parser is given. Finally, potential applications and future work are offered.

2. **Preprocessing**

The syntactic analysis is aided by a number of preprocessing steps included in the present system. First, the parser requires certain input formats. The system currently accepts UTF-8, CP 1256, or Romanized text, which then becomes Romanized if not already in that format, prior to syntactic analysis.

Secondly, the parser uses a dictionary file that lists words accompanied by their linkage requirements. A sizeable lexicon of Arabic is available from Buckwalter (2002), and is the basis of the lexicons for the present system. Romanized input is therefore written according to the transliteration used by Buckwalter.

Functional word categories are listed directly within the dictionary for the parser, since they consist of limited sets of closed class items. Words from open class categories are listed in separate files called up by the parser.

Additionally, a morphological engine is incorporated into the present system, as was done in the Persian link grammar system. The present system uses Buckwalter (2002) as a morphological analyzer for Arabic. Words from the input are segmented to constituent morphemes prior to any syntactic analysis. This system thus provides a tight coupling between morphology and syntax. This could give a more detailed description of Arabic structure than what is available presently.

Buckwalter's system supports many morphological phenomena in Arabic. Feature affixes on nominals, possessive morphemes, direct object enclitics, and verbal form affixes are included.

3. **Link Grammar**

The link grammar parser, originally developed for the English language (Sleator & Temperley 1991), provides a dependency-like method for parsing sentences. In this section, basic principles of link grammar are described and used to demonstrate how a grammar is developed. Afterwards, the parser is presented.

3.1 *Basics of link grammar*

In link grammar, each word has links that must be established with other words in a sentence. Examples of link types include a subject link that attaches a subject to a verb, and an object link that combines a transitive verb with its object.

Directionality and relative distance are the main principles upon which links in this grammar are defined and established. When two words have the same link type and their corresponding linking rules are in opposite directions (left, right), a link is established between them. Some words may have multiple links, some of which are more local to the word. According to the principle of distance, the sequence of linkage rules is equivalent to the sequence of application.

Links must also be established under the conditions of planarity, connectivity, satisfaction, and exclusion. Planarity states that links may not cross, connectivity holds that all the words of the sequence must be connected, satisfaction requires that links must satisfy the requirements of each individual word, and exclusion ensures that the same two words may only be connected by one link.

3.2 *Development of a link grammar*

In developing a link grammar, word types for a language are identified. These are specified by rules that establish links to, and thus

relationships with, other word types. The rules of the grammar can be complicated, involving multiple necessary links.

To understand a linkage rule, consider the relationship between a subject and its verb. For the case of a verbal sentence of Arabic, example (1) below shows linkage rules for a verb and a subject (since Romanized script is used, sentences are read from left to right):

(1)

Word	*Link type*	*Word*	*Link type*
Akl	S+	AHmd	S-
eat.perf.ms		Ahmad	

Both of these words have the same link type (S), which is a subject link. The verb Akl (أكل, 'to eat') has a '+' direction, which indicates establishment of a link towards the right. The proper noun AHmd can establish a link in the opposite direction. Since these words have the same link types with opposite directional requirements, a link may be established between them. A sample output of this is demonstrated in (2) below.

(2)

```
  +----S----+
  |         |
Akl       AHmd
```

A wide coverage link grammar for Arabic, based on such linkage rules, is now available to parse sequences of words in this language. The lexicon of words contains approximately 55,000 entries. The grammar currently contains about 50 rules, with approximately 40 parts of speech incorporated.

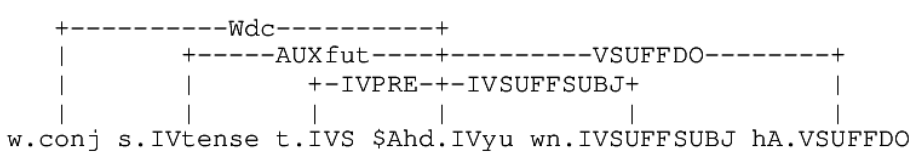
Since the present system also segments words into morphemes, linkage rules also may define the relationships between them. For example, a verbal stem may have multiple morphemic links for tense, features, and direct object enclitics. These links show how each morpheme relates to the verb stem, showing direct relationships between morphemes.

3.3 *A parser implementation*

The parser processes sentences based on the aforementioned linkage rules. It first reads in a link grammar dictionary that is separated into categories as described previously. Each word type is associated with certain linkage requirements which are used to parse word sequences to determine their grammaticality. The sentence along with its linked structure is then displayed.

The parser may be run in command line mode or through the web interface that is available. Sentences may be processed individually or in batch mode using a file of sentences. A user enters a sentence according to the required input formats and the system parses it morphologically and syntactically according to the linkage rules. The linked structure is displayed on the screen. Consider example (3) below:

(3) وستشاهدونها (Maamouri and Bies, 2004)



Relationships between individual items are shown. The majority of items here are morphemes, including a tense marker, subject affixes, a stem, and a direct object suffix. To understand this example, it is necessary to begin from the stem $Ahd (شاهد, 'to observe'), which is linked to four morphemes. First, phi-feature information is related to the stem by the prefixes t- and -wn, by the IVPRE (imperfective verb prefix) and IVSUFFSUBJ (imperfective verb suffix subject) link. These are the most local links to the verbal stem.

These feature affixes do not inhibit morphemes that are further away from attaching directly to the stem. The tense marker s is related to the stem $Ahd by the AUXfut (auxiliary future) link. In addition, the VSUFFDO (verb suffix direct object) links hA (ها, 'her') to the stem in a similar fashion. This ability to establish multiple linkages is extended to words in a sentence as well. Such descriptive abilities could be useful.

4. **Results**

Currently, the parser is able to handle Arabic sentences containing a variety of construction. So far the grammar assumes only VSO order, based on Standard Arabic. This structure is handled well by the parser, as is expected due to the flexibility shown in describing Persian SOV order and English SVO order.

The parser addresses morphological concerns, including verb formation and feature morphemes for nominals. *Iḍaafa* constructions, negation, relativized complementizers, clausal complements, conjunction, copular sentences, comparatives/superlatives, aspect markers, quantifiers and other structures have additionally been integrated. Sample parses (4) and (5) shown below demonstrate a variety of structures in Arabic.

(4) و رد الجيش الروسي على كل هذه الهجمات adapted from (Maamouri et al., 2003)

```
w rd Al jy$ Al rwsy ElY kl h'*h Al hjmAt

          +-----------------P----------------+
          +-----Sn----+------A------+        |        +------------QO------------+
   +--Wdc-+     +-Dal-+      +--Dal-+        +---PO---+          +---Ddp---+-Dal-+NSUFFFEMpl+
   |      |     |     |      |      |        |        |          |         |     |         |
w.conj rd.PVV Al.d jy$.Ndu Al.d rwsy.ADJ ElY.PREP kl.quant h'*h.DEMPRONF Al.d hjm.NAt At.NSUFFFEMPL
```

Here, the verb `rd` (رد, 'to respond') has both a subject (S) link with `jy$` (جيش, 'army') as well as a preposition (P) link to `ElY` (على, 'to'). This is similar to the example given above with morphemes, showing non-adjacent dependencies. An adjective (A) link is also shown. In addition, a quantifier object (QO) link is shown here is well.

(5) و لم يقدم المصدر أي إيضاحات حول أهداف الزيارة (Maamouri et al., 2003)

```
w lm y qdm Al mSdr >y <yDAH At Hwl >hdAf Al zyAr p

                    +--------------------------P---------------------+
   +---------Wdc---------+----------On---------+                      |
   |      +-----NEGi-----+-----Sn-----+        |                      |       +-------GEN-----+
   |      |      +-IVPRE-+     +--Dal-+        +---QO---+--NFEMpl--+       +---PO--+       +--Dal--+-NFEMsg--+
   |      |      |       |     |      |        |        |          |       |       |       |      |         |
w.conj lm.neg y.IV3MS qdm.IV Al.d mSdr.Ndu >y.quant <yDAH.NAt At.NFEMPL Hwl.PREP >hdAf.N Al.d zyAr.NapAt  p.NFEMSG
```

This example shows negation in the imperfect, directly linked to the root of the verb `qdm` (قدم, 'to offer'). Note that this verb has six links (prefix feature, negation, subject, conjunction, object, and preposition).

This illustrates how the root relates to both morphemes within the word it forms as well as to other words.

In addition, this example shows how *iḍaafa* constructions are handled, by the GEN (genitive) link. The possessed object `>hdAf` (أهداف, 'objectives') is directly related to the possessor `zyArp` (زيارة, 'trip') by this link. Longer *iḍaafas* are also handled.

Notice also that some of the link types are followed by lower case letters (e.g. `NEGi`). In link grammar, these are termed subscripts. They provide more in-depth descriptions of link types. For example, using the subscript `i` in the link `NEG` indicates that the negation `lm` may only link to imperfect verbs, thus avoiding the ungrammatical linkage of this negation element to a perfect verb.

Further issues that remain to be addressed include word order variations, topicalization, conditionals, and quotation embeddings. Quotation marks are currently discarded by the parser. Doing so could potentially be problematic for a parse of embedded citations, but so far has not been an issue.

5. **Ambiguity and Scoring**

Ambiguity is a critical issue as well since it may occur in various forms in Arabic. Al Daimi (2001:346-347) notes that "the issue of identifying ambiguities in Arabic language has been ignored in almost all the systems that attempted to process Arabic."

Ambiguity detection is possible using the link grammar system. As described above, all structures that are possible based on linkage rules are provided by the system. These display ambiguities in meaning present in sentences encountered. Rankings of these parses show which are the most feasible, according to a calculated cost vector.

The cost vector includes four components, which are `DIS`, `UNUSED`, `AND`, and `LEN`. First, a user may enter a cost with certain linkage by the use of square brackets; any number of sets may be used, and the more there are the greater the cost assigned to `DIS`. Any unused words (null links) in the parse are penalized, and this is reflected in `UNUSED`. The `AND` component applies to sentences with conjunctions. Linkages containing similar lengths of conjoined word-lists are preferred. The final component, `LEN`, prefers linkages which

have the least total length of links. Any parse containing constraints will be listed later in the ranking.

Example (6) shows a sententially ambiguous sentence is shown below. The input phrase is first shown, followed by the output from the system.

(6) غير احمد حسين

```
gyr >Hmd Hsyn

Found 3 linkages

  Linkage 1, cost vector = (UNUSED=0 DIS=1 AND=0 LEN=0)

   +---Sn---+-----G----+
   |        |          |
gyr.PV >Hmd.Nprop Hsyn.Nprop


  Linkage 2, cost vector = (UNUSED=0 DIS=2 AND=0 LEN=0)

   +---On---+-----G----+
   |        |          |
gyr.PV >Hmd.Nprop Hsyn.Nprop


  Linkage 3, cost vector = (UNUSED=0 DIS=2 AND=0 LEN=1)

   +--------On--------+
   +---Sn---+          |
   |        |          |
gyr.PV >Hmd.Nprop Hsyn.Nprop
```

The parser displays the number of possible parses, followed by each of the parses identified by linkage numbers as determined by the cost vector ranking. After the linkage number, the sentence is displayed with its structure as defined by the linkage grammar rules.

The grammar gives three possible linkages for this sentence. The first of these, which can be translated as ‘Ahmad Hussein changed’, is the highest ranked parse because it has the lowest cost vector (=1). The second linkage, ‘(He) changed Ahmad Hussein’, has a cost vector of 2, since subject links have lower penalized (`DIS`) cost than object links. The third linkage, ‘Ahmad changed Hussein’, has the highest cost vector (=3), because shorter links are preferred over longer ones.

At the moment, it is possible that parses from the system are not grammatical. An evaluation is needed to determine which rules are not restrictive enough.

6. **Discussion**

For purposes of comparison, it is important to identify the basics of any parsing system, along with problematic issues specific for Arabic. These are addressed here, along with an evaluation and an identification of advantages in the present work.

6.1 *Previous work*

Many constructions in the language are handled in previous Arabic parsers. These include transitive and intransitive sentences, copulars, noun constructs, word order variations in declarative sentences, wh- and yes/no questions, relative clauses, and embeddings. Numerical expressions, both in digit and written form, are handled by Al-Anzi (2001). Nominal and verbal sentences are included in previous work (Weinberg et al 1995; Abu-Arafah 1996; Othman et al. 2003; Shaalan 2005).

A couple of other issues are worth mentioning. A morphological module is considered important (Al Daimi & Abdel-Amir 1994; Ouersighni 2001; Othman et al. 2003). Robustness is essential in order to handle incorrect spellings or grammatical errors for purposes of tutoring (Weinberg et al. 1995).

Some issues are considered to be problematic in Arabic. Coordinating conjunctions are one problem (Abu-Arafah 1996; Ouersighni 2001). In addition, Ouersighni notes elliptical forms, negatives, complex determiners, ambiguity, anaphora, agreement and dependencies within a sentence, robustness, and text segmentation as problematic. This latter issue is critical for developing large scale parsers, due to long sentence lengths because of a less systematic punctuation system. Othman et al. (2003) notes this problem along with the lack of diacritics, free word order nature, and elliptical personal pronouns. For these reasons, Ouersighni notes the following

> "There is still no general language analyzer available for Arabic with sufficiently **wide coverage** to be sure that all expressions of the language are treatable with existing tools…Most systems select

> types of syntactic phenomena for treatment, with considerable lexical limitations." (Ouersighni 2001:1, emphasis original)

In addition, he notes that no present system is able to analyze actual Arabic texts, which are devoid of the diacritics or vowels in written form.

The system described by Ouersighni was the first to attempt to handle large scale parsing, taking on text segmentation through the use of strong separators. In addition, his system handles many of the difficult phenomena, including coordination, complex determiners, agreement and dependencies within a sentence, complements, and negative forms. Further issues that he feels need to be addressed include references outside the sentence, anaphoric references, robustness, recognition of idioms and composite words, and semantic elements.

6.2 *Comparison and evaluation*

The present system is able to handle many of the basic constructions of Arabic. Basic declarative sentences, copulars, noun constructs, relative clauses, and embeddings are built in. In addition, the system has a morphological component. Aspect markers, adverbs, and quantifiers are other important components of Arabic that are included in the present system; these don't appear to be mentioned in previous literature. The system is also able to handle some of the difficulties that were mentioned. These include coordination, complex determiners, and negative forms.

Certain issues from the literature are yet to be addressed in the present system. These include word order variations, numerical expressions, and questions. The system is capable of parsing long sentences, yet text segmentation might be a helpful asset. This could be implemented by identifying strong separators such as `f-` (ف) or `w` (و) and using the wall link that identifies sentence boundaries.

The present parser offers important advantages that need noting. Robustness is shown as words not in the lexicon are handled by the parser. It is able to guess a category for such words based on surrounding structural information. In addition, it is able to skip parts

of sentences that are ungrammatical, thus avoiding large processing times. This allows it to identify specific ungrammatical portions of sentences, while still being able to show a parse for the grammatical sections.

Identifying anaphoric references and dependencies both in and outside of the sentence could potentially be built in as well. As in the examples above, multiple linkages are possible for each word. It is critical to note that linkages can be established to individual morphemes in this system. Because of this, dependencies between a word referent and a morphemic anaphor may be established. This is only possible due to the fact that morphemes are separated, and thus are able to have their own linkage rules.

Ambiguity is another area that is addressed by the present system. The parser identifies and ranks ambiguous phrases and sentences through an exhaustive effort to produce all possible parses of a sentence.

Another important aspect of the parser is its speed; it is currently capable of processing about 1000 sentences per minute when using batch mode. A file of multiple sentences may be parsed relatively quickly.

7. **Applications and Future Work**

The system described herein has the potential to be used in a variety of applications. Some of these will now be described.

7.1 *Information extraction*

The link grammar framework is useful for purposes of information extraction. It has been integrated with a cognitive modeling system to extract semantic information from English text in the form of predicate logic. This could be applied to the present link grammar for Arabic to gather information from text.

7.2 *Grammar checking*

One possible use is in word processors as a grammar checker. Issues dealing with the checking of grammar for Arabic have been discussed by Shaalan (2005). He termed his approach syntax-based, as opposed to statistics-based or rule-based checking methods. This approach requires a lexicon, a morphological analyzer, and a parser. When no parse succeeds on a given text, it is considered incorrect.

This grammar checker was shown to be comparable with a commercially available one. Perhaps a link grammar implementation could be successful as well, as it would probably take a similar approach. In fact, recently an English link grammar has been implemented as a grammar checker in AbiWord[2], an open source word processor.

Feature checking for agreement between words (*e.g.* phi-features between verbs and nouns) would need to be extended in the present system in order for this to be feasible. This could be possible through the use of subscripts, which were introduced previously.

7.3 *Morphosyntactic information database*

Another potential use of the present system is in identifying and gathering interesting linguistic phenomena from large databases of written Arabic. Exporting the parses given from the link grammar parser to a Treebank formatted structure could potentially provide a more in-depth alternative to the current Arabic Treebank, since the current system provides descriptions of morphosyntactic structures.

A couple potential methods for doing this exist. Sleator & Temperley (1993) note that the grid of links output from the parser might be considered as a constituent framework. In the newest version of the parser, a system called the phrase parser enables a traditional constituent structure to be derived from a linkage. It does this by using a list of constituent types along with links that begin each one.

These displays can take a variety of forms, including a tree data structure. It should be possible to evaluate link grammar in terms of transferal into a constituency structure. Sleator & Temperley tested the

[2] http://www.abisource.com/

output from the English phrase parser with respect to the English Penn Treebank. On the complete text of the Penn Treebank, their parser correctly recognizes approximately 75% of the constituents[3].

We intend to more fully explore our Treebank formatted output with the use of this phrase parser. For Arabic, the links that begin constituent types would need reconsideration.

Schneider (1998) discusses some other possibilities. Schneider considers each link type in terms of head and dependent. He lists the types, along with which side should be the considered the head and which the dependent. Once a strict notion of a head for each type of link is found, conversion from a link grammar to dependency or constituency grammars, or even to semantic frameworks, could be possible.

8. **Conclusions**

Link grammar is capable of describing many distinct phenomena over a wide range of languages. An Arabic implementation of it has been developed for parsing. It has been shown that this parser is able to provide an in-depth morphosyntactic analysis of Arabic, as well as provide multiple parses of sentences that show ambiguity. Implementations for this parser could include use as a grammar checker or as a method for gathering linguistic phenomena from text corpora.

REFERENCES

Abu-Arafah, Adnan. 1996. *A Grammar for the Arabic Language Suitable for Machine Parsing and Automatic Text Generation*. Ph.D. dissertation, Illinois Institute of Technology.

Al-Anzi, Fawaz. 2001. “Sentential Count Rules for Arabic Language”. *Computers and the Humanities* 35:153-166.

Al Daimi, Khalid, & Abdel-Amir, M. 1994. “The Syntactic Analysis of Arabic by Machine”. *Computers and the Humanities* 28:29-37.

[3] http://www.link.cs.cmu.edu/link/ph-explanation.html

Al Daimi, Khalid. 2001. "Identifying Syntactic Ambiguities in Single-Parse Arabic Sentence". *Computers and the Humanities* 35:333-349.

Buckwalter, Tim. 2002. Buckwalter Arabic Morphological Analyzer Version 1.0. Linguistics Data Consortium, University of Pennsylvania. LDC Catalog No.: LDC2002L49.

Dehdari, Jon. & Lonsdale, Deryle. 2005. "A link grammar parser for Persian". In *First International Conference on Aspects of Iranian Linguistics*. Leipzig, Germany.

Liu, Carol. "Towards A Link Grammar for Chinese". Submitted for publication in *Computer Processing of Chinese and Oriental Languages - the Journal of the Chinese Language Computer Society*.

Maamouri, Mohamed, & Bies, Ann. 2004. "Developing an Arabic Treebank: Methods, guidelines, procedures, and tools". In *Proceedings of the 25th International Conference on Computational Linguistics* (COLING).

Maamouri, Mohamed, Bies, Ann, Jin, Hubert, & Buckwalter, Tim. 2003. Arabic Treebank: Part 1 v 2.0. Linguistics Data Consortium, University of Pennsylvania. LDC Catalog No.: LDC2003T06.

Othman, Eman, Shaalan, Khaled Fouad, & Rafea, Ahmed Abd El-Wahed. 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*, New Orleans, Louisiana. Available at: http://www-2cs.cum.edu/~alavie/semitic-MT-wshp.html

Ouersighni, Riadh. 2001. "A major offshoot of the DIINAR-MBC project: AraParse, a morphosyntactic analyzer for unvowelled Arabic texts". In *Proceedings of Arabic NLP Workshop at ACL/EACL*.

Protasov, Sergey. "A link grammar for Russian". Available at http://sz.ru/parser/Protasov-RussianLinkGrammar.pdf

Schneider, Gerold. 1998. *A linguistic comparison of constituency, dependency and link grammar*. Master's Thesis, University of Zurich.

Shaalan, Khaled. 2005. "Arabic GramCheck: a grammar checker for Arabic". *Software: Practice and Experience* 35: 643-665.

Sleator, Daniel, & Temperley, Davy. 1991. "Parsing English with a Link Grammar". Carnegie Mellon University Computer Science technical report CMU-CS-91-196.

Sleator, Daniel, & Temperley, Davy 1993. "Parsing English with a Link Grammar". In *Third International Workshop on Parsing Technologies*.

Weinberg, Amy, Garman, Joe, Martin, Jeffery & Merlo, Paola. 1995. "A Principle-Based Parser for Foreign Language Tutoring in German and Arabic". *Intelligent Language Tutors: Theory Shaping Technology* ed. by V. Melissa Holland, Jonathan Kaplan, & Michelle Sams, 23-44. Mahwah, N.J: Lawrence Erlbaum Associates.

Casbeer, Warren, Jon Dehdari and Deryle Lonsdale. 2007. "A link grammar parser for Arabic". In Perspectives on Arabic Linguistics: Papers from the annual symposium on Arabic linguistics. Volume XX: Kalamazoo, MI, USA, March 2006, Mughazy, Mustafa A. (ed.), 233–244. DOI: 10.1075/cilt.290

```
@inproceedings{casbeer-etal2007,
 author    = {Warren Casbeer and Jon Dehdari and Deryle Lonsdale},
 title     = {A Link Grammar Parser for {A}rabic},
 booktitle = {Perspectives on Arabic Linguistics},
 editor    = {Mustafa A. Mughazy},
 year      = {2007},
 volume    = {20},
 publisher = {John Benjamins},
 address   = {Kalamazoo, MI, USA},
 series    = {Current Issues in Linguistic Theory},
 pages     = {233--244},
 isbn      = {978-90-272-4805-3},
 doi       = {10.1075/cilt.290},
}
```