Designing an XML Lexicon Architecture for Arabic Machine Translation Based on Role and Reference Grammar

Yasser Salem and Brian Nolan

School of Informatics and Engineering
Institute of Technology Blanchardstown, Dublin, Ireland
E-mails: {firstname.surname}@itb.ie

Abstract

Role and Reference Grammar (RRG) is a functional theory of grammar. The main features of Role and Reference Grammar are the use of lexical decomposition, based upon predicate semantics, an analysis of clause structure and the use of a set of thematic roles organized into a hierarchy in which the highest-ranking roles are Actor (for the most active participant) and Undergoer. The theory allows a sentence in a specific language to be described in terms of its logical structure and grammatical procedures. The lexicon in RRG takes the position that lexical entries for verbs should contain unique information only, with as much information as possible derived from general lexical rules. We use the RRG theory to motivate the architecture of the lexicon. The lexicon is designed to reflect the word categories in the Arabic language with as much information as possible derived from general lexical rules. The lexicon stores the Arabic words in categories; each category is stored in an XML format datasource file. In order to be able to analyse Arabic by computer we must first extract the lexical properties of the Arabic words. Our system (UniArab) uses the lexicon to construct a logical structure for Arabic input sentences, also represented in XML, which is then used for generating the target language translation. We show the structure of the UniArab lexicon, discuss how it is used in the system, and show the user interface used for adding to the lexicon. The lexicon is built from individual words at present.

1 Introduction

This paper presents the development of a lexicon framework for Arabic language processing using the Role and Reference Grammar linguistic model. Machine translation is a sub-field of computational linguistics that investigates the use of computer software to translate text or speech from one natural language to another. The version of Arabic we consider in this paper is Modern Standard Arabic (MSA). When we mention Arabic throughout this paper we mean MSA which is distinct from classical Arabic (Alosh, 2005 & Schulz, 2005). In the Arabic linguistic tradition there is not a clear-cut, well-defined analysis of the inventory of parts of speech in Arabic. Attia (2008) mentioned that the traditional classification of Arabic parts of speech into nouns, verbs and particles is not sufficient for a complete computational grammar. This categorization, originally proposed by Sibawaih (Owens, 2006), remains the standard accepted scheme today. However, we have found it lacking when applied to machine translation (Salem et al., 2008b), and so, have developed our own lexical scheme. In an ideal Interlingua system lexical entries should be broken down into sets of semantic features (Hutchins and Somers, 1992). For example the word "man" is broken down into +human +male +adult. While this works in theory, in practice we cannot find enough semantic features to describe every entity in the world. For example "cow", "computer" and "chair" cannot be described using these sets of semantic features unless we invent a unique semantic feature for every object and this is practically impossible. Our system UniArab (Salem et al., 2008a) has been developed and is able to analyse Arabic sentences, and extract their logical structure. Through a detailed study of the Arabic language, we have been able to develop an analyser that incorporates many of the unique features and challenges present in Arabic. This logical structure is then used in the generation phase, where the sentence(s) is translated into another language, in this case, English. In this paper we describe the details of the XML-based metadata representing the RRG logical structure (XRRG) as a set of XML documents for each

component category of Arabic. We detail the various types of entries that encompass all Arabic words. We then introduce the main technologies used to support its development and discuss the user interface which allows for further addition to the lexicon.

2 The Role and Reference Grammar (RRG) Linguistic Model

Role and Reference Grammar (RRG) is a model of grammar (Van Valin 1997) that posits a direct mapping between the semantic representation of a sentence and its syntactic representation (Van Valin 2007). The theory allows a sentence in a specific language to be described in terms of its logical structure and grammatical procedures. RRG creates a linking relationship between syntax and semantics, and can account for how semantic representations are mapped into syntactic representations. We claim that RRG is very suitable for machine translation of Arabic via an Interlingua bridge implementation model. RRG is a mono strata-theory, positing only one level of syntactic representation, the actual form of the sentence and its linking algorithm can work in both directions from syntactic representation to semantic representation, or vice versa. In RRG, semantic decomposition of predicates and their semantic argument structures are represented as logical structures. The lexicon in RRG takes the position that lexical entries for verbs should contain unique information only, with as much information as possible derived from general lexical rules. The main features of RRG are the use of lexical decomposition, based upon predicate semantics, an analysis of clause structure and the use of a set of thematic roles organized into a hierarchy in which the highest-ranking roles are 'Actor' (for the most active participant) and 'Undergoer'. The RRG creates a relationship between syntax and semantics and can account for how semantic representations are mapped into syntactic representations. RRG also accounts for the very different process of mapping syntactic representations to semantic representations. Before developing the linking algorithms that govern these mappings, it is necessary to first introduce a general principle constraining these algorithms. Of the two directions, syntactic representation to semantic representation is the more difficult since it involves interpreting the morphosyntactic form of a sentence and inferring the semantic functions of the sentence from it. Accordingly, the linking rules must refer to the morphosyntactic features of the sentence. One question however remains; why should a grammar deal with linking from syntax to semantics at all. Simply specifying the possible realizations of a particular semantic representation should suffice. They refute this using the argument that theories of linguistic structure should be directly relatable to testable theories of language production and comprehension (Van Valin 2005).

3 The UniArab System

UniArab is a proof-of-concept system supporting the fundamental aspects of Arabic, such as the parts of speech, agreement and tenses. UniArab is based on the linking algorithm of RRG (syntax to semantics and vice versa). The conceptual structure of the UniArab system is shown in Figure 1. The UniArab system uses the lexicon to construct a logical structure for Arabic input sentences, also represented in XML, which is then used for generating the target language translation. UniArab stores all data in XML format. This data can then be queried, exported and serialized into any format the developer wishes. The system can understand the part of speech of a word, agreement features, number, gender and the word type. The syntactic parse unpacks the agreement features between elements of the Arabic sentence into a semantic representation (the logical structure) with the `state of affairs' of the sentence. In UniArab we have a strong analysis system that can extract all attributes from the words in a sentence. The structure of the UniArab system in Figure 1 breaks down into the following phases: **Phase (1) - Arabic language sentence:** The input to the system consists of one or more sentences in Arabic. **Phase (2) - Sentence Tokenizer:** Tokenization is the process of demarcating and classifying sections of a string of input characters. In this phase the system splits the text into sentence tokens. The resulting tokens are then passed to the word tokenizer phase. For example:

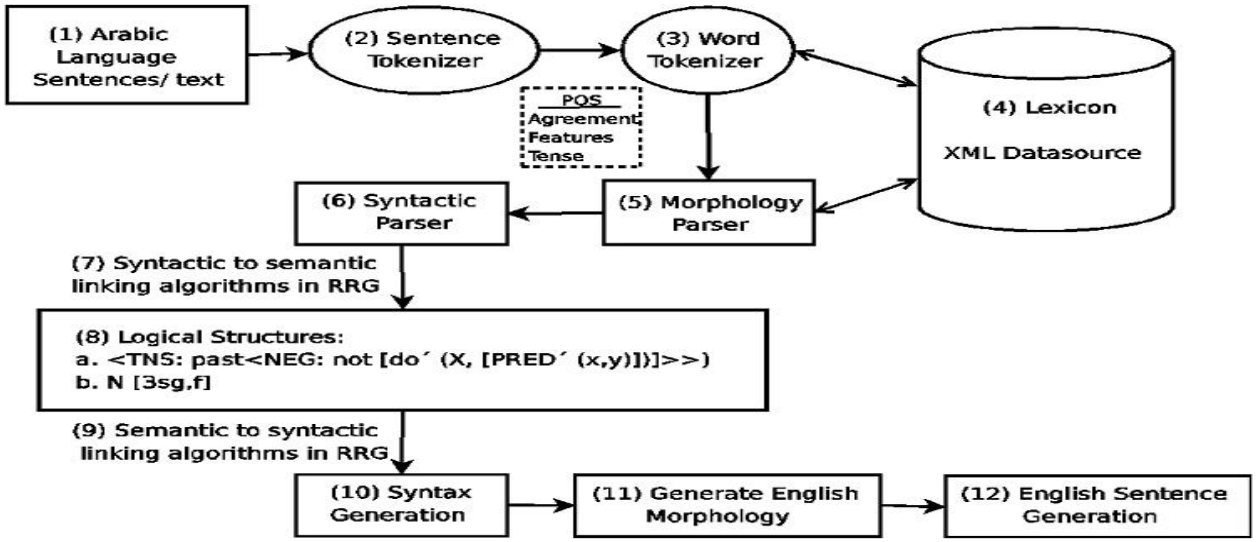

Figure 1: The conceptual architecture of the UniArab system

Figure 2: The Lexicon Interface of UniArab

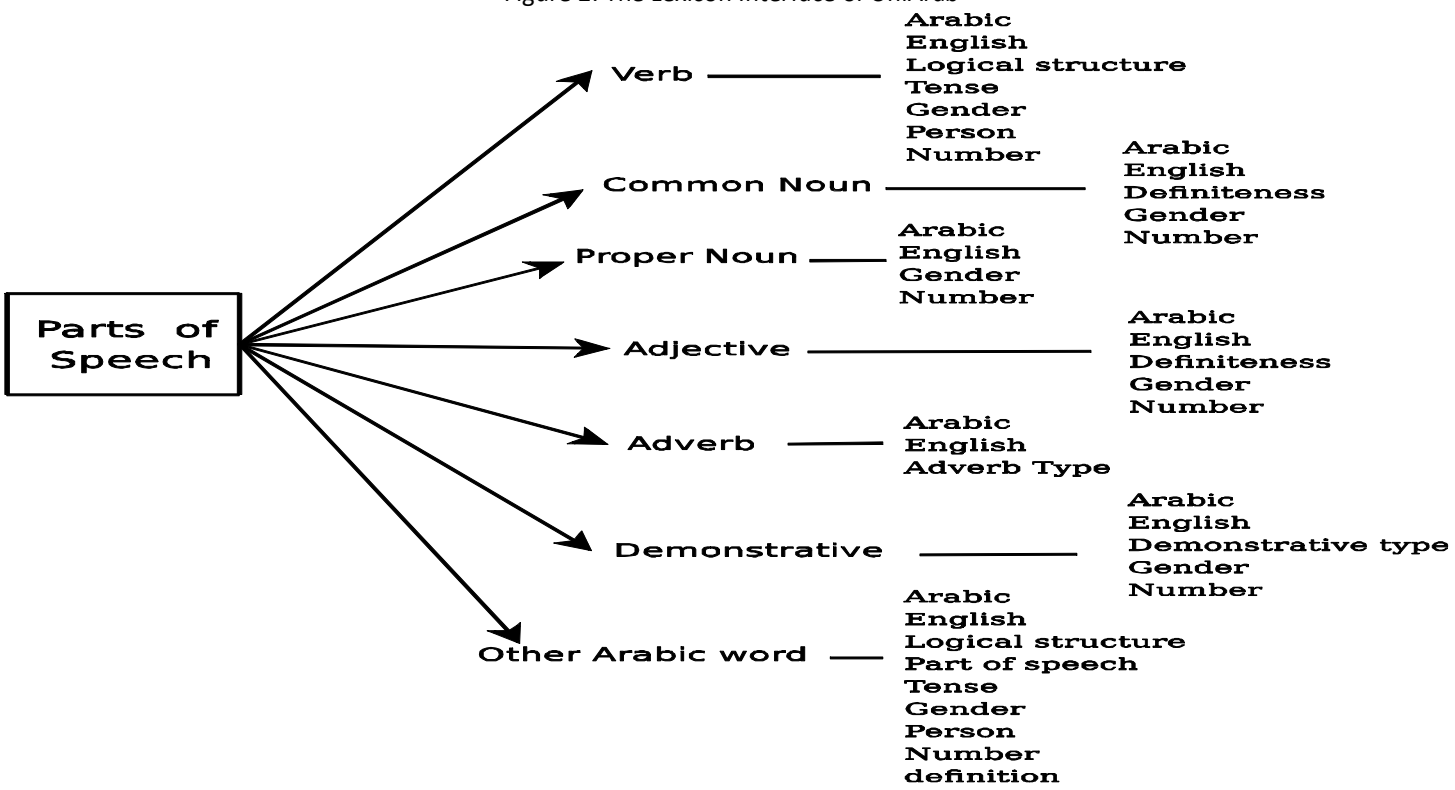

Figure 3: Information recorded in the UniArab Lexicon

قرأ خالد الكتاب. خالد تلميذ ذكي [1] qr'a ẖāld ālktāb. ẖāld tlmyḏ ḏky. will be two tokens; قرأ خالد الكتاب qr'a ẖāld ālktāb. and خالد تلميذ ذكي ẖāld tlmyḏ ḏky. The translation of these two sentences is *Khalid read the book. Khalid is a clever student.*

Phase (3) Word Tokenizer: There, sentences are split into tokens قرأ خالد الكتاب qr'a ẖāld ālktāb *Khalid read the book*, the output of phase 3 is as follows:

<sentence>
<word> قرأ qr'a </word>
<word> خالد ẖāld</word>
<word> الكتاب ālktāb </word>
</sentence>

Phase (4) Lexicon Data-source: A set of XML documents for each component category of Arabic. More details will be in section 4.

Phase (5) Morphology Parser: Directly works with both the Lexicon and Tokenizer to produce the word order. A connection is made to the data-source of phase 4, which has been implemented as a set of XML documents. The use of XML has the added advantage of portability. UniArab will effectively work the same regardless of the operating system. To understand the morphology of each word, we first tokenize each sentence and determine the word relationships. Phase 5 of the system holds all attributes specific to each word of the source sentence.

Phase (6) Syntactic Parser: Determines the precise phrasal structure and category of the Arabic sentence. At this point, the types and attributes of all words in the sentence are known.

Phase (7) Syntactic linking (RRG) We must first develop the link from syntax to semantics out of the phrasal structure created in Phase 6, if we are to create a logical structure that will generate a target language and also act as the link in the opposite direction from semantics to syntax. The system should answer the main question in this phase, **who does what to whom?** In this case the actor is *Khalid* and the undergoer is *the book*, as in: qr'a ẖāld ālktāb

Phase (8) Logical Structure: Creation of logical structure is the most crucial phase. An accurate representation of the logical structure of an Arabic sentence is the primary strength of UniArab. Below is a sample output from the UniArab system.

The Arabic equivalent of the past tense sentence قرأ خالد الكتاب qr'a ẖāld ālktāb *Khalid read the book*, is input as the source, and the following hold:

الكتاب ālktāb book:N خالد ẖāld Khalid:MsgN
قرأ qr'a read:V

The results of the parse can be seen in the following logical structure for the verb 'read'

<TNS:PAST[**do'**(x,[**read'**(x,(y)])]>

sg 3rd M PAST قرأ qr'a where : the Proper Noun is *Khalid* sg unspec M خالد ẖāld and the Noun is *the book* **sg def M** الكتاب ālktāb.

The output of phase 8 is as follow:

<TNS:PAST[**do'**(Khalid,[**read'**(Khalid,(book)])]>

Consider the following example, '*Omar is a student'* with a LS of **be'(Omar,[student'])**.

In Arabic this is: عمر تلميذ ` mr tlmyḏ. However, this presents a challenge since there is no verb 'to be' in Arabic, but this must be inferred for correct translation. Instead of saying *`Omar is a student'*, the Arabic equivalent would be *`Omar student*'. We also face the challenge of inferring the indefinite article, which does not exist in Arabic. All of the unique information for each word can thus be taken from the lexicon to aid in the creation of a logical structure of the target language.

Phase (9) Semantic to Syntax: Assuming we have an input and have produced a structured syntactic representation of it, the grammar can map this structure from a semantic representation. In this phase the system uses a linking algorithm provided by RRG, to determine actor and undergoer assignments, assign the core arguments and assign the predicate in the nucleus. The system uses semantic arguments of logical structures other than of the main verb.

Phase (10) Syntax Generation: This will be unique for each target language. In this phase the system uses the target language rule to generate the syntax. In this case English language rules are used.

Phase (11) Generate English Morphology: The system generates English morphology in an innovative way, generating the tenses not

[1] Arabic examples are written here by using Buckwalter Arabic Transliteration.

existent in Arabic but in English as well as the verb 'to be'. Verbs in English have a mood; e.g. indicative, subjunctive, imperative and can be in one of many tenses. On the other hand, verbs in Arabic have only three tenses. The solution is to recognize the difference between morphological features and syntactic functional categories. The tense features must be expressed analytically.

Phase (12) English Sentence Generation: The process of generating an English sentence can be as simple as keeping a list of rules. These rules can be extended through the life of the MT system. The system will use some operations in English such as vowel change: examples; man men. Sometimes this accompanies affixations: break broke broken **(=broke + en)**.

4 An XML-based lexicon

In this section we describe the details of the XML-based metadata representing the RRG logical structure (X-RRG) in Phase (4) *Lexicon Datasource* as a set of XML documents for each component category of Arabic.

In order to build this system and represent the data sources, we use the XML language and Java. The most recent recommendation of the XML language has been presented by (Bray et al., 2008). XML has become the default standard for data exchange among heterogeneous data sources (Arciniegas, 2000). The UniArab system allows data to be stored in XML format. This data can then be queried, exported and serialized into any format the developer wishes. We choose to create our data source as XML, for optimum support or different platforms. It was also easier as we used Arabic letters not Unicode inside the data source. XML fully supports Arabic. We created our search engine using Java.

4.1 Advantages of XML

XML is a generalized way to store data, which is not married to any particular technology. This makes it easy to store something, and then come back and retrieve it later with some other technology and process. Using XML to exchange information offers a number of advantages, including the following:

Easily build: A well formed data element must be enclosed between tags. The XML document can be parsed without prior knowledge of the tags. XML allows you to define all sorts of tags with all sorts of rules, such as tags representing data description or data relationships.

Human readable: Using intelligible tag names will make it possible to read, even by novices.

Machine-readable: XML was designed to be easy for computers to process. XML is completely compatible with Java and portable. Any application can process XML on any platform, as it is a platform-independent language.

XML fully supports Arabic: We chose to create our data-source as XML files, for optimum support of different platforms. It was also easier as we used Arabic letters rather than Unicode inside the data-source.

XML search engine: It is easy to extend the search sample to display more information about the search. Search by Java API Document Object Model (DOM) is the ideal tool for searching collections of XML documents.

4.2 Lexicon interface

In order to allow for robust user interaction with the lexicon, we use a graphical interface to capture the information for each part of speech. The user selects the part of speech of the word to be added, and is then presented with only the attributes relevant to the selected part of speech. The interface also limits the user's selections to acceptable values and ensures that all attributes are filled. With this technique, we minimize the risk of human errors, and therefore the information is more accurate. The graphical interface is quicker and easier when a user adds a new word in the lexical (XML data source). Figure 2 shows the entry interface that is implemented as part of the UniArab system.

4.3 Lexical representation in UniArab

Lexical frames represent the language-dependent lexicon. We use an XML data source to represent the UniArab lexicon. The lexicon creates pointers to corresponding conceptual frames or attributes of each word. These frames also have relations which link them to verb class frames, which are organized hierarchically according to the particular language. Since the verb is the key component when analysing using RRG, each verb has an associated logical structure, which is later used to determine the

logical structure of the full sentence. The verb attributes, in particular, are of great importance in correctly extracting sentence logical structure further down the processing chain, helping to answer the basic question 'Who does what to whom?' In free word order sentences, for example, يحب قيس ليلى yhb qys lylā, multiple orders are possible including verb-subject-object, verb-object-subject or subject-verb-object. The attributes of the verb define the gender of the subject. Given the masculine gender of the verb in this case, the Syntactic Parser will look for a masculine proper noun to make the actor for this sentence. If there is more than one masculine proper noun in such a case, then Modern Standard Arabic defines the first proper noun as the actor. The Morphology Parser will be extended so that it can deal with words that are defined in multiple categories, deciding which should be processed. Meanwhile the Syntactic Parser, so far, has only been implemented for extracting word order, though it will be extended to deal with word ambiguities in future versions.

4.4 Lexical properties

Figure 3 shows the structure of the Lexicon including the properties stored for each word category. For all categories, an Arabic word is stored along with its English representation. Since word ambiguity has not been dealt with so far, there is a one to one mapping for the simple sentences which UniArab processes up to now. However, word ambiguity is supported in the structure, with each possible case stored as a separate record. All search results will be passed to the Morphology Parser to decide which is taken.

Since the **verb** is the key component when analysing using RRG, each verb has an associated logical structure, which is later used to determine the logical structure of the full sentence. The tense of the verb is also stored within its metadata along with the person. The verb type also stores the gender, which in Arabic must be either masculine or feminine; there is no natural gender. The number property in Arabic can be singular, dual or plural. These properties help the Syntactic Parser analyse the sentence, since there must be agreement with the subject and verb, among other rules.

In Tables 1, we show an example of records for verb in the Lexicon. The English translations of these verbs are `*read'*. An example of the XML record for a verb in the Lexicon is shown here in (1), following.

(1) XML record for a verb in the Lexicon

```
<قرأ
    EnglishTranslate="read"
    LogicalStructures=
    "<TNS:PAST[do'(x,[read'(x,y])]>;"
    NumberVerb="sg"
    P.O.S="Verb"
    genderVerb="M"
    personVerb="3rd"
    tenseVerb="PAST"
/>
```

For the **Common Noun**, we also store information for the gender and number, but we also include a definitiveness attribute which indicated with the noun in Arabic is definite or indefinite. The definitive article, ال āl, in Arabic is attached to the beginning of a word to make it definitive. Table 2 shows examples of two different Arabic noun words, whose English translations are `*trees*' and `*book*'. Please note that `book' is def +.

Proper nouns in Arabic are not capitalized (there is no capital letter in Arabic), and so they are defined within the lexicon. They have the same attributes mentioned previously. Table 3 shows examples of two different Arabic proper noun words, whose English translations are `Omar' and `Eman'.

Adjectives also have the same properties. Table 4 shows examples of two different Arabic adjective words, whose English translations are `*short*' and `*long*', Note that `*long*' is def+.

For **demonstratives**, number and gender remain, and we introduce a type attribute. In the Arabic language, this can have one of three values: near to the speaker, far from the speaker or between near and far from the speaker. Table 5 shows examples of two different Arabic demonstratives, whose English translations are `*this*' and `*that*'.

Arabic verb	قرأ qr’a
English translation	read
Logical structure	[do'(x,[read'(x,(y)])]
Tense	past
Gender	m
Person	3rd
Number	singular

Table 1: Verb

Arabic Noun	أشجار ‘ašǧār	الكتاب ālktāb
English translation	trees	the book
Definiteness	indefinite	definite
Gender	f	m
Number	plural	singular

Table 2: Noun

Arabic Noun	عمر ‘mr	إيمان ‘iymān
English translation	Omar	Eman
Gender	m	f
Number	singular	singular

Table 3: Proper Noun

Arabic verb	قصير qṣyr	الطويلة ālṭwylh
English translation	short	the long
Definiteness	indefinite	definite
Gender	m	f
Number	singular	singular

Table 4: Adjective

Arabic Adverb	بجانب bǧānb	اليوم ālywm
English translation	beside	today
Adverb type	place	time

Table 5: Demonstrative

Arabic verb	هذا hḏā	ذلك ḏlk
English translation	this	that
Demonstrative type	close	far
Gender	m	m
Number	singular	singular

Table 6: Adverb

Arabic other words	و w	هي hy
English translation	and	she
Logical structure	NON	NON
Part of speech	conjunction	pronoun
Tense	NON	NON
Gender	NON	f
Person	NON	3rd
Number	NON	singular
Definition		

Table 7: other Arabic words

Adverbs in Arabic do not have a gender or number, and so the only property stored in the Lexicon is the type. The Type refers to time or place, time such as `today' or `tomorrow' and places like `under', `in', or `on' etc. Table 6 shows examples of two different Arabic adverbs, whose English translations are `beside' and `today'.

We have created a final category, ‘**Other Word’**, to store words that cannot be categorized into one of the above. For this category, we have allowed all attributes to be defined, since we cannot determine which apply in advance. Table 7 shows examples of two different Arabic *Other words*, whose English translations are `and' and `she'.

5 Results

In this paper, we are concerned with discussing the Lexicon, and so have not presented detailed results (Salem & Nolan 2009). However, in Table 8,[2] we show an example with a non-standard word order.

[2] Note that Arabic sentences should be read from right to left

Arabic	يحب ليلى قيس yḥb lylā qys
Human-translated	Qays loves Laila.
Google	Leila loves measured
Microsoft	Love laili Qais
UniArab	Qays loves Laila.

Table 8: Free word order (verb noun noun first possibility)

The output of the Google translator (Google, 2009) is faulty in the actor, the system can not analyse who does what, the actor is Qais but the output makes the object the subject. Microsoft's translator (Microsoft, 2009) translates each word while ignores the word order and the meaning of sentence. It also makes the object the subject. UniArab successfully translates the sentence in its entirety. since it is able to use the word attributes stored in the Lexicon to correctly identify the actor.

Arabic	يحب قيس ليلى yḥb qys lylā
Human-translated	Qays loves Laila.
Google	Qais likes of Laila
Microsoft	Love Qais laili
UniArab	Qays loves Laila.

Table 9: Free word order (verb noun noun second possibility)

In Table 9, the output of the Google translator is faulty in the verb meaning and the system added `of' without any meaning in this sentence. Microsoft's MT translated each word while ignoring the word order and meaning of the sentence. UniArab successfully translates the sentence in its entirety.

Arabic	قيس يحب ليلى qys yḥb lylā
Human-translated	Qays loves Laila.
Google	Qais likes of Laila
Microsoft	Qais love laili
UniArab	Qays loves Laila.

Table 10: Free word order (verb noun noun third possibility)

Table 10, shows the third possible sentence order. The output of the Google translator is faulty in verb meaning and adds an extra `of' without any meaning. Microsoft's MT translates each word while ignoring the word order, tense and meaning of sentence. UniArab successfully translates the sentence in its entirety.

6 Summary and future work

In this paper we have presented a detailed account of the lexical properties of Arabic parts of speech and the attributes for each type of word and how these are implemented within the UniArab Lexicon. A thorough Lexicon allows UniArab to provide more accurate translations when compared with automated translators from Google and Microsoft though they have a much wider coverage than UniArab at present. It is clear that with more complicated sentence structures, the attributes of the various words are essential in correctly translating. Storing these attributes within the Lexicon allows for the processing stages discussed to be implemented. We would like to extend this work by addressing the question of ambiguity. We feel that RRG is suited to overcoming word ambiguity by using sentence structure, and would like to explore this further. We would also like to explore the auto generation of lexicon information from Arabic three-letter source verbs as a way to quickly populate the lexical source. The main focus remains to increase the coverage of the system, adding further, more complex verbs, and dealing with the associated challenges of larger sentences.

7 References

Alosh, Mahdi. (2005). *Using Arabic: A Guide to Contemporary Usage*. Cambridge: Cambridge University Press.

Arciniegas, F., (2000). *XML Developer's Guide*. McGraw-Hill Companies.

Attia, Mohammed A. (2008). Handling Arabic Morphological and Syntactic Ambiguity within the LFG Framework with a View to Machine Translation. Ph.D. thesis.

Bray, T., Paoli, J., Sperberg-McQueen, C. M., aler, E., and Yergeau, F., (2008). *Extensible arkup Language (XML) 1.0 (Fifth Edition)*.

Google, (2009). Google translator. http://translate.google.com.

Hutchins, J. and Somers, H. L., (1992). *An introduction to machine translation*. Academic Press.

Microsoft, (2009).Microsoft translator. http://www.windowslivetranslator.com/Default.aspx.

Salem, Yasser., and Brian Nolan. (2009). UNIARAB: An Universal Machine Translator System For Arabic Based On Role And Reference Grammar, in *Proceedings of the 31st Annual Meeting of the Linguistics Association of Germany (DGfS 2009)*, University of Osnabruck, Germany.

Salem, Y., Hensman, A., and Nolan, B., (2008a). Implementing Arabic-to-English Machine Translation using the Role and Reference Grammar Linguistic Model. In *Proceedings of the Eighth Annual International Conference on Information Technology and Telecommunication (IT&T 2008)*, Galway, Ireland, October 2008.

Salem, Y., Hensman, A., and Nolan, B., May (2008b). Towards Arabic to English machine translation. In *ITB Journal Issue Number 17*.

Schulz. Eckehard (2005). *A Student Grammar of Modern Standard Arabic*. Cambridge: Cambridge University Press.

Owens, Jonathan. (2006). A Linguistic History of Arabic. Oxford University Press.

Van Valin, Robert D. 2007. The Role and Reference Grammar analysis of three place. *CEEOL Contemporary Linguistics*, **63**:31–63.

Van Valin, Robert D. (2005). *Exploring the Syntax-Semantics Interface*. Cambridge: Cambridge University Press.

Van Valin, Robert D. and LaPolla, R. (1997). *Syntax: Structure, Meaning, and Function*. Cambridge: Cambridge University Press.