BASMA: BibAlex Standard Arabic Morphological Analyzer

Sameh Alansary

Director of Arabic Computational Linguistics Centre, Bibliotheca Alexandrina
Head of Phonetics and Linguistics Department, Faculty of Arts, Alexandria University
sameh.alansary@bibalex.org

Abstract: ***Arabic morphology poses special challenges to computational natural language processing systems. Its rich morphology and the highly complex word formation process of roots and patterns make computational approaches to Arabic very challenging. Morphological analyzers are preprocessors for text analysis. This paper sheds the light on BASMA-Tool (BibAlex Standard Arabic Morphological Analyzer) that has been initiated at Bibliotheca Alexandrina (BA). The BASMA tool is based on Buckwalter Arabic Morphological Analyzer (BAMA). It focuses on fixing its problems, adding a set of useful morphological features that BAMA does not provide, and disambiguating its multiple solutions. This is done depending on a well training data and a hybrid system (Rule based and memory based). Precision and Recall are the evaluation measures used to evaluate BASMA tool. At this point, precision measurement was 93.37% while recall measurement was 96.9%. The percentages are expected to rise by implementing the improvements while working on larger amounts of data.***

1 INTRODUCTION

Arabic is a language of rich morphologycompared to other languages especially Europeanlanguages. It is based on both derivational and inflectional morphology. The richness of Arabic morphology makes the analysis process difficult to deal with. On the one hand, morphological analysis process is used in most of the NLP applications such as information retrieval, spell checking and machine translation. On the other hand, morphological analysis is the first step before syntactic analysis. Furthermore, it is anessential step in semantic analysis [1].

Arabic has a high degree of ambiguity resulting from its diacritic-optional writing system and common deviation from spelling standards (e.g., Alif and Ya variants)[2].

Morphological analysis for text corpora is a prerequisite for many text analytics applications, which has attracted many researchers from different disciplines such as linguistics (computational and corpus linguistics), artificial intelligence, and natural language processing, to morphosyntactically analyze text of different languages including Arabic. Recently, several researchers have investigated different approaches to morphological and syntactic analysis for Arabic text. Many systems have been developed which vary in complexity from light stemmers, root extraction systems, lemmatizers, complex morphological analyzers, part-of-speech taggers and parsers [3].

In 2007, Bibliotheca Alexandrina (BA) has started an important project of building the “International Corpus of Arabic (ICA)”. It is a serious attempt to build a representative Arabic corpus as being used all over the Arab world that is able to support research on Arabic. It is planned to contain 100 million words morphologically, syntactically and semantically analyzed. The first stage of linguistic analysis of the International corpus of Arabic is to analyze the 100 million words of the ICA corpus morphologically [4][5][6].

The stem-based approach “concatenative approach” has been adopted as a linguistic approach to analyze the ICA morphologically. There are many morphological analyzers for Arabic; some of them are available for research and evaluation while the rest are proprietary commercial applications. Buckwalter Morphological analyzer (BAMA) is one of the well-known analyzers in the literature and has even been considered the “most respected lexical resource of its kind” [6].It is designed as a main database of word forms interacting with other concatenation databases. In Buckwalter, every word is entered separately, and the stem is used as the base form of a word. Words are viewed as being composed of basic units that can combine with morphemes governed by morphotactic rules; thus, Buckwalter Morphological Analyzer entails the use of three lexicons: a Prefixes Lexicon, a Stem Lexicon, and a Suffixes Lexicon.

Section 2 of this paper will discuss the trials that use BAMA in the morphological disambiguation process. Section 3 will review the BibAlex Standard Morphological Analyzer system and why there was a need to enhance BAMA, through explaining and discussing some of the main problems noticed in its output. This section will also introduce to what extent it is different from BAMA (2004). Moreover, section 4 will show the current state of the development and BASMA`s results and section 5 includes a comparison between BASMA and MADA. Finally, section 6 will state the conclusion.

2 RELATED WORK

MSA morphological analysis, disambiguation, part-of-speech (POS) tagging, tokenization, lemmatization and diacritization have received a lot of focus; for an overview, see [7]. And more recently, there has been growing body of work on Dialectical Arabic (DA) [8], [9], [10] and [11] among others. In this paper, the discussion will be focused on two systems that are commonly used by researchers in Arabic NLP: MADA [12], [13], [14] and [11] and AMIRA [15].

The primary purpose of Morphological Analysis and Disambiguation for Arabic (MADA3.2) is to extract as much linguistic information as possible about each word in the text, from given raw Arabic text, in order to reduce or eliminate any ambiguity concerning the word. MADA uses ALMORGEANA (an Arabic lexeme-based morphology analyzer) to generate every possible interpretation of each input word. It then applies a number of language models to determine which analysis is the most probable for each word, given the word's context.

MADA uses up to 19 orthogonal features in order to choose, for each word, a proper analysis from a list of potential analyses derived from the Buckwalter Arabic Morphological Analyzer (BAMA) [16]. The BAMA analysis that most closely matches the collection of weighted, predicted features is chosen. The 19 features include 14 morphological features that MADA predicts using 14 distinct Support Vector Machines (SVMs) trained on the PATB. The other five features that MADA capture information such as spelling variations and n-gram statistics.

Since MADA selects a complete analysis from BAMA, all decisions regarding morphological ambiguity, lexical ambiguity, tokenization, diacritization and POS tagging in any possible POS tag set are made in single action [11], [17], and [18]. The choices are ranked in terms of their score. MADA has over 96% accuracy on basic morphological choice (including tokenization, but excluding case, mood, and nunation) and on lemmatization. MADA has over 86% accuracy in predicting full diacritization (including case and mood). More detailed comparative evaluations can be found in [12], [17] and [13].

The AMIRA toolkit includes a tokenizer, a part of speech tagger (POS), and a base phrase chunker (BPC), also known as a shallow syntactic parser. The technology used in AMIRA is completely different from that of MADA, since it is based on supervised learning with no explicit dependence on knowledge of deep morphology, it relies on surface data to learn generalizations.

AMIRA was enhanced, in later versions, with a morphological analyzer and a named-entity recognition (NER) component. Moreover, both tools are similar in using a unified framework that postpones each of the component problems as a classification problem to be solved sequentially. AMIRA adopts a multi-step approach to tokenization, part-of-speech tagging and lemmatization, in contrast to MADA that handles all of these and more in a single action. The analysis that MADA provides is deeper than that of AMIRA, namely by identifying syntactic case, mood and construct state in the morphological tag, however, it is slower in processing. In addition, AMIRA provides additional utilities - BPC and NER - that are not supported by MADA. Both tools are somewhat brittle, academic prototypes implemented in Perl; they rely on third-party software utilities which the end-user must install andconfigure separately [2].

3 BIBALEX STANDARD ARABIC MORPHOLOGICAL ANALYZER (BASMA)

Initially, Buckwalter Arabic Morphological Analyzer (BAMA) has been selected, since it was the most suitable lexical resource to our approach [4]. Although it has many advantages including its ability to provide a sufficient amount of information such as Lemma, Vocalization, Part of Speech (POS), Gloss, Prefix(s), Stem, Word class, Suffix(s), Number, Gender, Definiteness and Case or Mood, it does not always provide all the information the ICA requires, and in some cases, the provided analyses would need some modification. The obtained results may vary between giving the right solution for the Arabic input word, provide more than one result that needs to be disambiguated to reach the best solution, provide many solutions, but none of them is right, segment the input words wrongly without taking the segmentation rules in consideration or provide no solutions. Consequently, solutions enhancement would be needed in these situations.

Number, gender and definiteness need to be modified according to their morphosyntactic properties. Some tags had been added to the ICA lexicon, some lemmas and glossaries had been modified and others had been added. In addition, new analysis and qualifiers had been added as root, stem pattern and name entities [5].

The process of developing a morphological analyzer tool for ICA began in 2007 which is known as BibAex Arabic Morphological Analyzer Enhancer (BAMAE). It is a system that has been built to morphologically analyze and disambiguate the Arabic texts depending on BAMA's output. It was preferred to use BAMA's enhanced output of ICA, since it contains more information than any other BAMA's enhanced systems. And this is the reason why the members of ICA team aimed to build their own morphological analyzer tool.

In order to reach the best solution for the input word, BAMAE preforms automatic disambiguation process carried on three levels, depends primarily on the basic POS information (Prefix(s), Stem, Tag and Suffixes) obtained from enhanced BAMA's output. [5], [6]:

- Word level which avoids or eliminates the impossible solutions that Buckwalter provides due to the wrong concatenations of prefix(s), stem and suffix(s).

- Context level where some linguistic rules have been extracted from the training data to help in disambiguating words depending on their context.
- Memory based level which is not applicable in all cases; it is only applicable when all the previous levels failed to decide the best solution for the Arabic input word.

After selecting the best POS solution for each word, BAMAE detects the rest of information accordingly. It detects the lemmas, roots (depending primarily on the lemmas), stem patterns (depending on stems, roots and lemmas), number (depending on basic POS and stem patterns), gender (depending also on basic POS, stem patterns and sometimes depending on number), definiteness (depending on POS or their sequences), case (depending on definiteness and sequences of POS) and finally it detects the vocalization of each word.

Figure 1 shows BAMAE architecture starting from the input text and the numerous solutions for each word in order to predict the best POS solution for each word and then detect the rest of information accordingly.

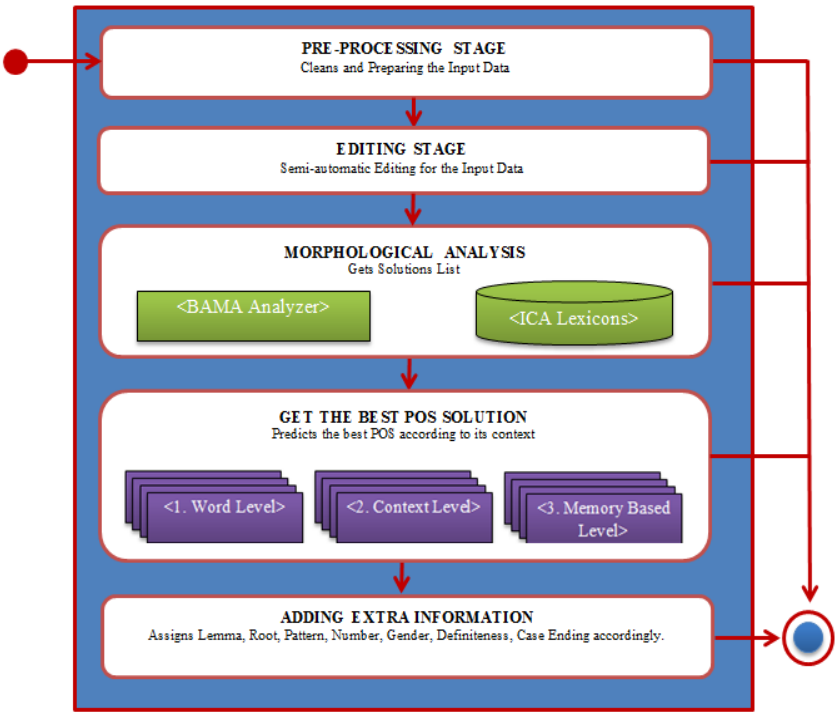

Figure 1: BAMAE Architecture.

The precision measurement of BAMAE was 92% while recall measurement was 89%. These percentages were expected to be raised by implementing the improvements while working on larger amounts of data [6].

After discovering BAMA's output problems, and handling these problems in the BAMAE, the decision was to handle these problems in BAMA. But, not all of BAMA's output problems have been handled in BAMA. Others have been handled by implementing Arabic linguistic rules, depending on the kind of the problem. Handling these problems required some modifications in the Perl code of BAMA (AraMorph). Moreover, more development was needed such as a new feature that Buckwalter does not provide, was added to BAMA's lexicons namely stem pattern as well as another feature that is found in lexicons, but does not appear in BAMA's output solutions namely root. By handling these problems and revamping some functions in BAMAE another update has been released known as BASMA. The following sub-sections review how these problems have been handled and implemented in BASMA:

A. Problems handled in BAMA's lexicons:

As mentioned before, not all problems are necessarily handled in this stage, it only handles problems that are related to the lack in grammar-lexis specifications, uncovered concatenations of some words, uncovered prefixes or suffixes in Arabic, wrong segmentations, wrong lemmas, wrong roots and wrong tags. These problems have been fixed in BAMAs' lexicons and/or their compatibility tables[1] according to the problem type.

The problems that are related to the lack in grammar-lexis specifications, uncovered prefixes or suffixes in Arabic and wrong tags have been fixed in both BAMAs' lexicons and their compatibility tables, because if a new prefix, tag or suffix is added, some constrains must be added to rule which combinations of these prefixes, tags and suffixes are linguistically acceptable and which are not, depending on the nature of Arabic language. In addition, the lack in grammar-lexis requires adding more constrains to avoid the wrong combinations that BAMA does not constrain. Figure 2 shows an example for the problem of detecting wrong tags and lack in grammar-lexis specifications for some words and how it has been handled in this stage.

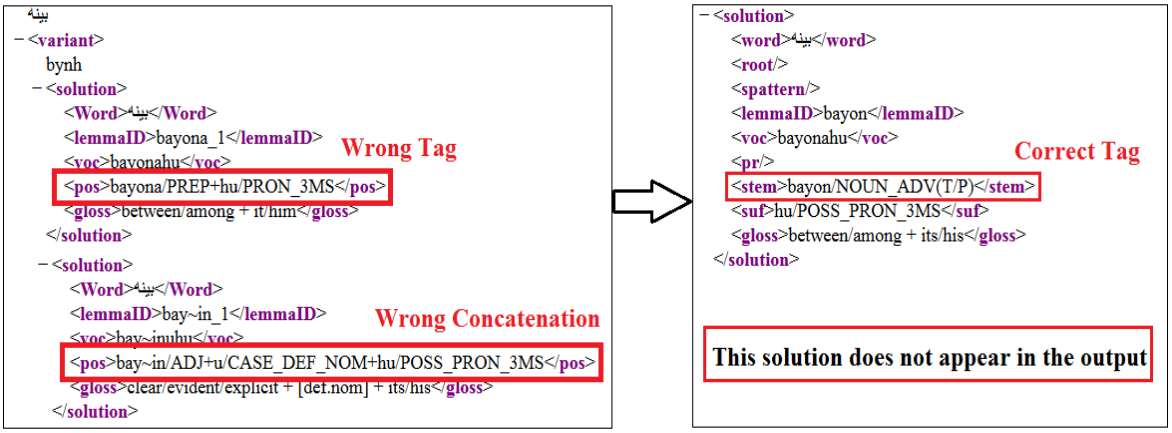

Figure 2: Example for wrong tags and concatenations.

The problems that are related to wrong lemmas or roots and wrong or new glosses have been handled in BAMA's lexicons and specifically in dicStems lexicon without being handled in the compatibility tables. As mentioned before the root feature does not appear in BAMA's output, although it is found in the dicStems lexicon. Moreover, unfortunately not all of the roots that are available founded in this lexicon are Arabic roots, so there has to be some modifications in these roots. After reviewing all roots in the dicStems lexicon, they are displayed in the output.

Although the stem pattern is not used in BAMA's lexicon at all, it is found that the stem pattern feature is very useful in enriching the lexicons, we have depended on it in the disambiguation process of ICA texts. The stem patterns have been detected automatically, depending on root and stem of some words and depending on root, lemma and stem in other words. Then, these stem patterns have been added and mapped in the dicStems lexicon. Figure 3 shows an example for the problem of wrong lemmas and roots and how the roots and stem patterns appear now in the output:

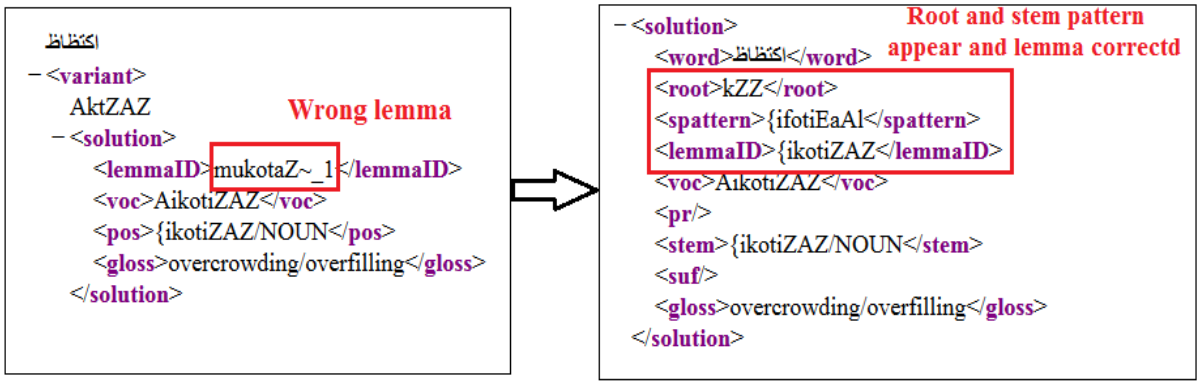

Figure 3: Example for wrong lemma and displayed root and stem pattern in the output.

Some words did not have any solutions for one of three reasons. First, some words are not analyzed altogether by BAMA; second, some words are analyzed, but none of the provided solutions is suitable to their contexts in the text; third, some words are wrongly segmented by BAMA [5] and [6]. Such words have been inserted in BAMA's dicStems lexicon with it suitableconstrains to generate it correctly. An example of the second category of unanalyzed words is the passive form

of the word 'حرموا''be forbidden/be deprived'. After inserting the suitable transliteration, stem, tag (with suitable constraints) and gloss for this word, it is analyzed correctly as figure 4 shows:

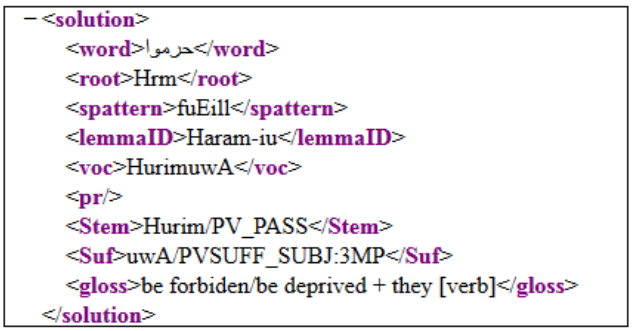

Figure 4: Example of recently inserted word.

It must be noted that after handling the problem of wrong concatenations and the lack in grammar-lexis specifications, there will be no need to handle this part in BASMA. Furthermore, these modifications are still in progress to enhance the input solution source for BASMA as much as possible, hence enhancing the morphological analysis results.

B. Problems handled by Arabic Linguistic rules:

The problems that have been handled through the linguistic rules are the problems that are related to morphosyntactic properties such as number, gender, definiteness and case ending. There are some linguistic rules that have been extracted from the previously analyzed data to help in assigning the right solution in the next time the data is analyzed. The assigning process may depend on basic POS and stem pattern of each best selected solution such as in number and gender, or it may depend either on the POS of each best selected solution or the POS of the surrounding words in addition to its POS such as definiteness and case (context based). These features are no longer displayed in BAMA's output, since the correctness of detecting the number and gender by the linguistic rules in BASMA is more adequate. In addition, both definiteness and case ending features are context based features, so there is no need to display such information in the solutions list while selecting the best morphological analysis for each word.
Figure 5 shows some words that BAMA has assigned the wrong number and gender to them, and how these words have been handled in BASMA.

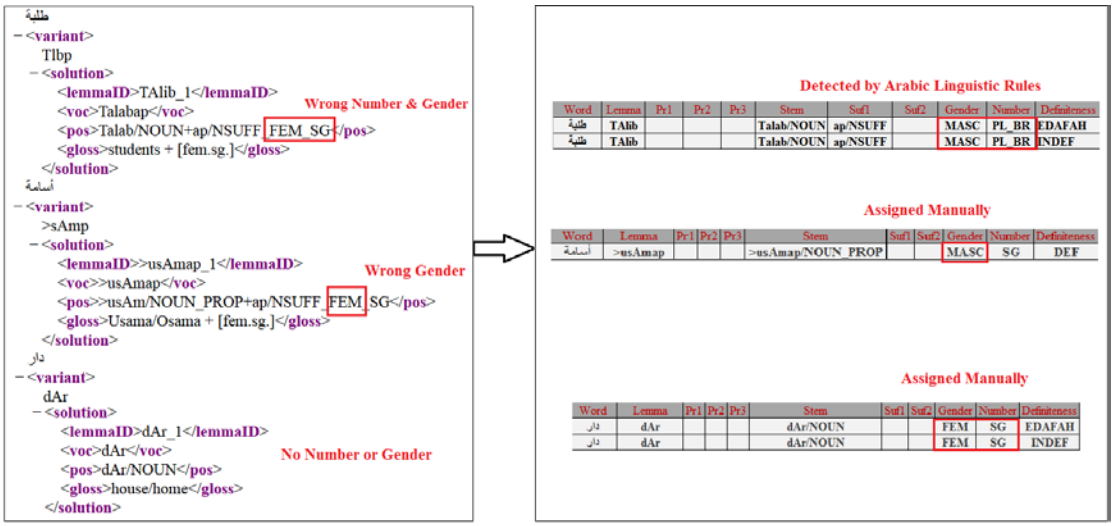

Figure 5: Example for the corrected gender and number features.

It must be noted that in order to prevent such features from appearing in BAMA's output some handling have been done in dicSuffixes BAMA's lexicon. All information that refers to any of these features have been deleted. The accuracy of rules in detecting gender and number are acceptable and can be enhanced, while the accuracy of rules in detecting definiteness and case ending still needs more modifications, since these features need more syntactic information.

C. Needed modifications in BAMA's AraMorph Perl file:

There are some modifications that are needed in BAMA's AraMorph Perl file. These modifications need to be compatible with the new added features in BAMA's output; root and pattern. In addition, there are some needed modifications to make the parsing process of BAMAs' solutions in BASMA easier. These modifications are 1) separating the prefixes and suffixes from the stem, 2) displaying the input word of every word, and 3) showing the x_solution of BAMA with only the words that have no solutions at all. Figure 6 shows BAMA's output solutions after these modifications.

Figure 6: BAMA's output after modifications.

4 RESULTS AND EVALUATION

To evaluate BASMA, a blind test data set (1,000,000 representative words) was run using BASMA, and results were compared to a manually annotated version. Precision, recall and accuracy are the evaluation measures used to evaluate the BASMA system. Precision is a measure of the ability of a system to present only relevant results. Recall is a measure of the ability of a system to present all relevant results. The evaluation has been conducted on two levels; the first level includes the precision, recall and accuracy for each qualifier separately as table 1 shows. The second level includes the basic POS in addition to adding a new qualifier each time to investigative how it would affect the accuracy as table 2 shows.

TABLE 1
PRECISION, RECALL AND ACCURACY FOR QUALIFIERS SEPARATELY

Qualifier	Precision	Recall	Accuracy
Lemma	97.16	99.95	97.07
Pr1	98.50	99.90	97.00
Pr2	99.90	99.96	99.80
Pr3	100	100	100
Stems	96.83	99.95	93.67
Tags	96.39	99.96	92.78
Suf1	96.27	99.25	95.82
Suf2	99.86	99.97	99.72
Gender	98.46	99.87	97.74
Number	98.84	99.78	97.67
Definiteness	93.94	98.51	87.89
Root	99.30	99.80	98.60
Stem Pattern	97.80	99.80	95.60

TABLE 2
ACCURACY DECREASING AS A RESULT OF ADDING NEW QUALIFIER EACH TIME TO THE MAIN POS TAG

POS + Qualifiers	Accuracy
Prefix(s) + Stem + Tag + Suffix(s)	93.37
Prefix(s) + Stem + Tag + Suffix(s) + *Lemma*	93.11
Prefix(s) + Stem + Tag + Suffix(s) + *Lemma + Root*	92.95
Prefix(s) + Stem + Tag + Suffix(s) + *Lemma + Root + Pattern*	92.95
Prefix(s) + Stem + Tag + Suffix(s) + *Lemma + Root + Pattern + Number*	92.41
Prefix(s) + Stem + Tag + Suffix(s) + *Lemma + Root + Pattern + Number+ Gender*	92.03
Prefix(s) + Stem + Tag + Suffix(s) + *Lemma + Root + Pattern + Gender + Number + Definiteness*	88.10

Finally, precision measurement was 93.37% while recall measurement was 96.9%. The percentages are expected to increase by implementing the improvements while working on larger amounts of data. Figure 7 shows an example of some features of BASMA's results.

Word	lemmaid	pr1	pr2	pr3	stem	suf1	suf2	gen	num	def	root	stem_pattern
/D					BOF_Doc							
/T					BOF_Tit							
الجيوب	Eayob	Al/DET			Eayuwb/NOUN			MASC	PL_BR	DEF	Eyb	fuEuwl
الفنية	fan~iy~	Al/DET			fan~iy~/ADJ	ap/NSUFF		FEM	SG	DEF	fnn	faEoliy~
..					Punc							
سبب	sabab				sabab/NOUN			MASC	SG	EDAFAH	sbb	faEal
كارثة	kAriwap				kAriv/NOUN	ap/NSUFF		FEM	SG	EDAFAH	krv	faAEil
الأنابيب	>unobuwb	Al/DET			>anAbiyb/NOUN			MASC	PL_BR	DEF		
T/					EOF_Tit							
/P					BOF_Prg							
تحقيق	taHoqiyq				taHoqiyq/NOUN			MASC	SG	EDAFAH	Hqq	tafoEiyl
محمد	muHam~ad				muHam~ad/NOUN_PROP			MASC	SG	DEF	Hmd	mufaE~al
هندي	hinodiy~				hinodiy~/NOUN_PROP			MASC	SG	DEF		
P/					EOF_Prg							
/P					BOF_Prg							
المشع	jaSaE	Al/DET			jaSaE/NOUN			MASC	SG	DEF	jSE	faEal
..					Punc							
ونقص	naqoS	wa/CONJ			naqoS/NOUN			MASC	SG	EDAFAH	nqS	faEol
الكميات	kam~iy~ap	Al/DET			kam~iy~/NOUN	At/NSUFF		FEM	PL	DEF	kmm	faEoliy~
المأساوية	maToruwH	Al/DET			maToruwH/ADJ	ap/NSUFF		FEM	SG	DEF	TrH	mafoEuwl
هما	humA				humA/PRON			MASC	DU	DEF		
المتهمان	mut~aham	Al/DET			mut~aham/NOUN	Ani/NSUFF		MASC	DU	DEF	thm/whm	mufotaEal
الرئيسيان	ra}iysiy~	Al/DET			ra}iysiy~/ADJ	Ani/NSUFF		MASC	DU	DEF	r's	faEiyliy~
في	fiy				fiy/PREP							
أزمة	>azomap				>azom/NOUN	ap/NSUFF		FEM	SG	EDAFAH	'zm	faEol
أنابيب	>unobuwb				>anAbiyb/NOUN			MASC	PL_BR	EDAFAH		
البوتاجاز	buwtAjAz	Al/DET			buwtAjAz/NOUN			MASC	SG	DEF		
التي	Al~atiy				Al~atiy/REL_PRON			FEM	SG	DEF		
وصفها	waSaf~i				waSaf/PV	a/PVSUFF_SI	hA/PVSUFF_I				wSf	faEal
كثير	kaviyr				kaviyr/NOUN_PROP			MASC	SG	DEF	kvr	faEiyl
من	min				min/PREP							
المواطنين	muwATin	Al/DET			muwATin/NOUN	iyna/NSUFF		MASC	PL	DEF	wTn	mufaAEil
بأنها	>an~a	bi/PREP			>an~a/SUB_CONJ	hA/PRON_3F						
مقلقة	mufotaEal				mufotaEal/NOUN	ap/NSUFF		FEM	SG	EDAFAH	fEl	mufotaEal
بعد	baEod				baEod/NOUN			MASC	SG	EDAFAH	bEd	faEol
أن	>an~a				>an~a/SUB_CONJ							

Figure 7: BASMA output results.

5 COMPARING BASMA WITH MADA

MADA (Morphological Analysis and Disambiguation for Arabic) is selected to be compared with BAMAE since both of them use Buckwalter's output analyses to help in disambiguating the Arabic texts. The primary purpose of MADA 3.2 is to extract as much linguistic information as possible about each word in the text, from given raw Arabic text, in order to reduce or eliminate any ambiguity concerning the word. MADA does this by using ALMORGEANA (an Arabic lexeme-based morphology analyzer) to generate every possible interpretation of each input word. MADA then applies a number of language models to determine which analysis is the most probable for each word, given the word's context.

In order to compare between BASMA and MADA, a text; to be used to evaluate both systems, was selected from ICA training data to facilitate the comparing process. To make the comparing process more accurate some modifications have been done in MADA's format to be compatible with BASMA's format. For example, in the number qualifier the feature of singular (s) was modified to be (SG), in the case qualifier the feature of nominative (u) was modified to be (NOM), in the tags qualifier the verbs were handled with relation to aspect and stem category. The comparing process will be done among some qualifiers; diacritization, tags, stems, number, gender and definiteness including Arabic words only as Table 3 shows:

TABLE 3
COMPARING RESULTS BETWEEN BASMA AND MADA

Qualifier	BASMA	MADA
Diacritization	91.11	78.78
Tags	95.94	85.28
Stems	97.08	91.34
Number	99.10	64.93
Gender	99.12	66.67
Definiteness	97.53	60.61

There are some notes that must be taken into consideration:

- The problems of detecting the diacritization in BAMAE are related to either the wrong prediction of the case ending or wrong prediction of the whole solution.
- The problems of detecting the diacritization in MADA are related to the wrong prediction of the case ending, wrong prediction of the whole solution, missing some diacritics in some words, or missing all diacritics in some words.
- The problems of detecting the tags in MADA are related to either the wrong prediction of the tags or the differences in some tags from BASMA. For example the adverbs of time or place in BASMA are assigned with 'NOUN_ADV(T)' or 'NOUN_ADV(P)', while they are assigned with 'NOUN', sub conjunction 'SUB_CONJ', and preposition 'PREP' in MADA. This happens as a result of using BAMA's output without enhancing these tags. In addition, the wrong concatenations of BAMA's output causes problems in detecting some tags.

- The problems of detecting stems in both BASMA and MADA are related to the wrong prediction of the solution.
- The problem of detecting number, gender and definiteness in MADA are related to using BAMA's output without regarding the morphosyntactic properties.
- The cases in BASMA and MADA can't be compared, since MADA assigns case without regarding the diacritics of the case. For example, it assigns the accusative case 'ACC' for both 'a/ACC' and 'i/ACC' which are differentiated in BASMA.
- There are some qualifiers in BASMA which are not used in MADA; Root and Stem Pattern. The root qualifier has been assigned with accuracy 99.45% while the stem pattern qualifier has been assigned with accuracy 96.34%.
- The lemma qualifier has been assigned in BASMA with accuracy 97.64%, while it is not used in MADA.

6 CONCLUSIONS

About 20 million words have been disambiguated using (BASMA). The evaluation has been done using precision and recall measurements for 1,000,000 words. Precision measurement was 93.37% while recall measurement was 96.9%. The percentages are expected to increase by implementing the improvements while working on larger amounts of data. If the analysis tools reach a deadlock and cannot improve any more enhancements, the data will be corrected manually.

REFERENCES

[1] M. Gridach& N. Chenfour, Developing a new system for Arabic morphological analysis and generation. In Proceedings of the 2nd Workshop on South and Southeast Asian Natural Language Processing (WSSANLP'11) (pp. 52-57), November 2011.

[2] A. Pasha, A. Mohamed, M. Diab, A. El Kholy, R. Eskander, N. Habash , M. Pooleery, O. Rambow& R. Rohth, Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of Arabic. In Proceedings of the Language Resources and Evaluation Conference (LREC), Reykjavik, Iceland, 2004.

[3] M. Sawalha, E. Atwell & M. Abushariah, SALMA: Standard Arabic Language Morphological Analysis. In Communications, Signal Processing, and their Applications (ICCSPA) 1st International Conference on (pp. 1-6). IEEE, February 2013.

[4] S. Alansary, M. Nagi & N. Adly, Towards Analysing the International Corpus of Arabic (ICA): Progress of Morphological Stage. In Proceedings of 8th International Conference on Language Engineering, Egypt, 2008.

[5] S. Alansary. 2012, BAMAE: Buckwalter Arabic Morphological Analyser Enhancer. In Proceedings of 4th international conference on Arabic language processing, Mohamed Vth University Souissi, Rebate, Morocco, May 2-3, 2012.

[6] S. Alansary & M. Nagi, The International Corpus of Arabic: Compilation, Analysis and Evaluation. ANLP August 2014.

[7] N. Habash, Introduction to Arabic Natural Language Processing. Morgan & Claypool Publishers, 2010.

[8] R. Al-Sabbagh & R. Girju, A supervised POS tagger for written Arabic social networking corpora. In Jancsary, J., editor, Proceedings of KONVENS, pages 39–52. ÖGAI, September. Main track: oral presentations, 2012.

[9] E. Mohamed, B. Mohit & K. Oflazer, Annotating and Learning Morphological Segmentation of Egyptian Colloquial Arabic. In Proceedings of the Language Resources and Evaluation Conference (LREC), Istanbul, 2012.

[10] N. Habash, R. Eskander, & A. Hawwari. A Morphological Analyzer for Egyptian Arabic. In NAACL-HLT Workshop on Computational Morphology and Phonology (SIGMORPHON), pages 1–9, 2012.

[11] N Habash, R. Roth, O. Rambow, R. Eskander, & N. Tomeh, Morphological Analysis and Disambiguation for Dialectal Arabic. In Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), Atlanta, GA, 2013.

[12] N. Habash& O. Rambow, Arabic Tokenization, Part-of-Speech Tagging and Morphological Disambiguation in One Fell Swoop. In Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05), pages 573–580, Ann Arbor, Michigan, 2005

[13] R. Roth, O. Rambow, N. Habash, M. Diab& C. Rudin, Arabic morphological tagging, diacritization, and lemmatization using lexeme models and feature ranking. In proceedings of the Conference of the Association for Computational Linguistics (ACL); Companion Volume, Short Papers, Columbus, Ohio, June. Association for Computational Linguistics, 2008 .

[14] N. Habash, O. Rambow& R. Roth, MADA+TOKAN: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization. In Proceedings of the 2nd International Conference on Arabic Language Resources and Tools. The MEDAR Consortium, April, 2009.

[15] M. Diab, K. Hacioglu, & D. Jurafsky, Automated Methods for Processing Arabic Text: From Tokenization to Base Phrase Chunking. In proceedings of Arabic Computational Morphology: Knowledge-based and Empirical Methods. Kluwer/Springer, 2007.

[16] T. Buckwalter, Buckwalter Arabic Morphological Analyzer Version 2.0. Linguistic Data Consortium (LDC) catalogue number LDC2004L02, ISBN 1-58563-324-0, 2004.

[17] N. Habash& O. Rambow, Arabic diacritization through full morphological tagging, In the Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers (pp. 53-56). Association for Computational Linguistics, 2007.
[18] R. Roth, O. Rambow, N. Habash, M. Diab& C. Rudin, Arabic Morphological Tagging, Diacritization, and Lemmatization Using Lexeme Models and Feature Ranking. In Proceedings of ACL-08: HLT, Short Papers (Companion Volume), pages (117–120), Columbus, Ohio, USA, June 2008.

BIOGRAPHY

Dr. Sameh Alansary

He is professor of computational linguistics in the Department of Phonetics and Linguistics and the head of Phonetics and Linguistics Department, Faculty of Arts, Alexandria University. He obtained his MA in Building Arabic Lexical Databases in 1996, and his PhD from Nijmegen University, the Netherlands in building a formal grammar for parsing Arabic structures in 2002. His main areas of interest are concerned with corpus work, morphological analysis and generation, and building formal grammars.
He is also the head of Arabic Computational Linguistics Center in Bibliotheca Alexandrina. He is supervising and managing the Universal Networking Language project in Library of Alexandria since 1-6-2005 till now.
Dr. Alansary is the co-founder of the Arabic Language Technology Center (ALTEC), an NGO aims at providing Arabic Language resources and building a road map for Arabic Language Technology in Egypt and in the Middle East. He has many scientific works in Arabic Natural Language Processing published in international conferences and periodicals, and a member in many scientific organizations: (1) Egyptian Society of Language Engineering, Cairo, (2) Arabic Linguistic Society - USA, (3) Association of Computational Linguistics - USA – Europe, (4) Universal Networking Language foundation, United Nations, Geneva, Switzerland.

محلل مكتبة الأسكندرية الصرفي للعربية المعاصرة (BASMA)

سامح الأنصاري

مدير مركز اللغويات الحاسوبية العربية – مكتبة الأسكندرية

قسم الصوتيات واللسانيات، كلية الآداب، جامعة الأسكندرية

sameh.alansary@bibalex.org

ملخص—يعد الصرف العربي أحد التحديات الأساسية في الأنظمة المستخدمة في المعالجة الآلية للغة العربية. فالعربية غنية بالكثير من التنوعات والتعقيدات الصرفية حيث نجد أنه من الجذر الواحد يمكن توليد العديد من الكلمات المختلفة في الوزن الصرفي. تركز هذه الورقة الضوء على أحد المحللات الصرفية الآلية الذي تم بناؤه في مكتبة الأسكندرية (المحلل الصرفي للغة العربية المعاصرة لمكتبة الأسكندرية). وهذا المحلل يقوم بتحليل الكلمات تبعا لتواردها في سياقات مختلفة بالاعتماد على التحليلات الصرفية الواردة من المحلل الصرفي الشهير تيم باك ولتر. فيقوم هذا المحلل بمعالجة المشاكل الواردة من باك وولتر، كما يعتمد في عملية فك اللبس الصرفي على نظام هجين يعتمد على بعض القواعد اللغوية وبعض النماذج اللغوية الإحصائية المستخلصة من عينة لغوية، وهذه العينة اللغوية عبارة عن مجموعة نصوص محللة تحليلا صرفيا، وقد وصلت نسبة الصحة في هذا المحلل الصرفي إلى 93.37% حيث استطاع المحلل التعرف على 96.9% من التحليلات الصرفية للكلمات. ومن المتوقع أن تزيد هذه النسبة بتطبيق مزيد من التحسينات على ذلك المحلل.