Volume 9, No. 10, October 2018
Journal of Global Research in Computer Science
REVIEW ARTICLE
Available Online at www.jgrcs.info

TIMELINE OF THE DEVELOPMENT OF ARABIC POS TAGGERS AND MORPHOLOGICALANALYSERS

Mohammad Elsheikh Salim Elsheikh [*1] and Eric Steven Atwell[2]

*[1]Computer Science department, Shendi University, Shendi, Sudan
mohammad_elgarrai@ush.sd[1]
[2] School of Computing, University of Leeds, Leeds, UK
E.S.Atwell@leeds.ac.uk[2]

Abstract: Over the past two decades, since around 2000, Arabic NLP researchers have investigated a variety of approaches to PoS-tagging and morphological analysis. In this paper, we present this research as a timeline or a list of events in chronological order, to illustrate the evolutionary development of the field.We present a timeline of 24 different approaches and tools for Arabic Part of Speech (POS) tagging and morphological analysis. Most of the work focuses on Modern Standard Arabic (MSA). A few systems aim at Dialect Arabic (DA); the Classical Arabic (CA) gets the least attention.

Keywords: Part of Speech Tagging, morphological analysis, Modern Standard, Dialect, Classical Arabic

INTRODUCTION

The Arabic language is rich resource for natural language processing research. Arabic is characterized by a rich and complex morphology or word-structure, so that development of morphological analyzers and Part-of-Speech taggers has been considerably more challenging for Arabic than for English, and many different methods and tools have developed and evolved over the past two decades. We present a timeline of 24 different approaches and tools for Arabic Part of Speech (POS) tagging and morphological analysis. Most of the work focuses on Modern Standard Arabic (MSA). A few systems aim at Dialect Arabic (DA); the Classical Arabic (CA) gets the least attention.

Classical Arabic (CA) is the language which appeared in the Arabian Peninsula centuries before the emergence of Islam and continued to be the standard language until the medieval times. CA continues to the present day as the language of religious teaching, poetry, and scholarly literature.

Modern Standard Arabic (MSA) is a descendent variant of CA and is used today throughout the Arab World in writing and in formal speaking. [1]The modern form of Arabic is called Modern Standard Arabic (MSA) and it is the form used by all Arabic-speaking countries in publications, the media and academic institutions. It is the language universally understood by Arabic speakers around the world. MSA is spoken by people from different Arab countries where the local dialect may not be mutually intelligible. MSA is a simplified form of Classical Arabic, and follows its grammar. The main differences between CA and MSA are that MSA has a much larger (more modern) vocabulary, and does not use some of the more complicated forms of grammar found in Classical Arabic [1][2][3][4].

Dialectal Arabic (DA) refers to the day-to-day native vernaculars spoken in different parts of the Arab World. Dialectal Arabic is related to MSA but different from MSA phonologically, morphologically and lexically and has no standardized orthography or written form – Arabic dialect speakers are taught at school to write in MSA [3][5].

Morphological analysis is used to analyze the word andidentify its morphemes; by doing that we can assign the correct part-of-speech tag to the word, such as noun, verb, particle. A PoS tagger is a system that uses context to assign tag to words. The goal is to assign each word in a text a word class (part of speech / tag).Automatic text tagging (part-of-speech tagging) is an important first step in discovering the linguistic structure of large text corpora. It is the basic and primary analysis step in many natural language processing applications; and it is a very important resource for finding certain patterns in a language, analyzing word frequencies, syntactical structures and other analysis. Part-of-speech information facilitates higher-level analysis, such as recognizing noun phrases and other patterns in text[6][7][8].

There are three general approaches to deal with the tagging problem:

1. Rule-based approach: consists of developing a knowledge base of rules written by linguists to define precisely how and where to assign the various POS tags[2][9].
2. Statistical approach: consists of building a trainable model and to use previously-tagged corpus to estimate its parameters. Once this is done, the model can be used to automatically tagging other texts[2][9].
3. Hybrid approach: Consists in combining rule-based approach with a statistical one[2][4][9].

Over the past two decades, since around 2000, Arabic NLP researchers have investigated a variety of approaches to PoS-tagging and morphological analysis. In the following, we present this research as a timeline or a list of events in chronological order, to illustrate the evolutionary development of the field.

TIMELINE OVER TWO DECADES: 24 APPROACHES

(MORPH2, 2000) [10][11] [33]

MORPH2 is based on a knowledge-based computational method implemented in an oriented-object framework using Java programming language, and uses an XML-based scheme for annotation. And it is a new version of the Arabic Morphological analyzer MORPH1. This MORPH2 involves five steps; tokenization, preprocessing, affixal analysis, morphological analysis, vocalization & validation. The new add steps are Vocalization & validation. It has been

evaluated in the same corpus that consists of a collection of various Arabic texts and contain 51404 words. The accuracy rate was 89.77% for the recall measure and 82.51% for the precision measure. This tool is embedded in Multiple systems such as DECORA (i.e. A system of agreement error detection and correction for non vocalized Arabic texts), MASPAR (i.e. A multi-agent system for parsing Arabic), QASAL (i.e. A Question Answering System for Arabic Language), The anaphora annotating system of Mezghani.

(Morpho3, 2000) [11][10]

Morpho3, is the successor for Morpho2, has been developed in RDI labs. Morpho3 deal with all the possible Arabic words and removes the need to be tied to a fixed vocabulary. It uses a powerful dynamic m-gram statistical disambiguation technique. This system covers almost the whole Arabic morphological phenomenon. It can deal with MSA text as well as CA text. Also, this morphological model is a simple compact one. So, scripting the morphological entities in Morpho3 is an easy straightforward task as long as one knows the classical morphological properties of the entities. It has been evaluated in two aspect coverage and disambiguation using different size (15000 words, 50000 words, 250000 words, 500000 words) of corpus. The coverage rate was as fallow 98% for 15000 words, 98.73% for 50000 words, 98.9% for 250000 words, and 99.05% for 500000 words. The Disambiguation was as fallow 70.21% for 15000 words, 79.34% for 50000 words, 88% for 250000 words, and 94.3% for 500000 words.

(Ouersighni, 2001) [12][13]

AraParse system (Arabic Parser), a morpho-syntactic analyzer for unvowelled Modern Standard Arabic texts, it developed and extended from spelling checker version. The AraParse is composed of several modules, the GenLex system to generate lexicon from the DIINAR.1 lexical database, morphological module, and the parser is created from a grammar by means of the AGFL GenParser tool. Taking into account the lexicons database size was increases from 3.5 Mb for the spelling checker to 13 Mb. Every input sentence goes the follows steps in AraParse, first the text is divided into sentences, then each sentence is then segmented into lexical units, after that the output of the morpholexical module is used as the input for the parsing module by means of an AGFL sub-grammar based internal interface. A set of morpho-syntactic information necessary for syntactic analysis is associated with the lexical entries. The actual syntactic analysis is carried out by the parser which gives the syntactic structures in labeled tree or bracket form.

(Khoja, 2001)[2]

He describes a hybrid Arabic part-of-speech tagger for MSA that used -Statistical tagger with Viterbi Algorithm and rule-based techniques. Initial they began with tagset that had five main tags (Noun, Verb, Particle, Residual, Punctuation), and extended it to 35 tags then to 131 tags taking into account clitics and subcategories that is derived from traditional Arabic grammatical theory. For training and testing the tagger 4 corpus are used. For training a corpus of 50,000 words extract from the Saudi Al-Jazirah newspaper, date 03/03/1999. An 90% accuracy rate is achieved when tested in The remaining 59,040 words of the Saudi ``Al-Jazirah", 3,104 words of the Egyptian ``Al-Ahram" newspaper, date 25/01/2000, 5,811 words of the Qatari ``Al-Bayan" newspaper, date 25/01/2000, And 17,204 words of Al-Mishkat an Egyptian published paper in social science, April 1999.

(Beesley, 2001) [7][14][15]

Beesley redesigned, rebuilt and developed his morphological analyzer-generator Xerox (root-based morphology), based on dictionaries from an earlier project at ALPNET, using finite-state technology. The system online based that analyzes orthographical words that may include full, partial, or no diacritics; and if diacritics are present, they automatically constrain the ambiguity of the output. The Arabic entry page is mostly filled up with a Java applet that displays a virtual Arabic keyboard. As words are typed, the appropriate Unicode Arabic characters are added to an internal buffer. When the user enters a word a special Perl CGI script send it to the morphological analyzer, which is implemented as a finite-state transducer (FST) and give Several output strings with possible analysis of the input word. Each solution is passed to a morphological generator FST, which is exactly the same as the analyzer except for having a lower-level language that is restricted to fully-voweled strings. The various solutions are also tokenized into morphemes, which are looked up in a dictionary of English glosses. And sent back to the user's Internet browser for display into an HTML page with fully-voweled strings, and the English glosses.

The system underlying contains of 4 part of speech tags, a lexicons include about 4930 roots and pattern dictionary with about 400 phonologically. In practice, the average root participates in about 18 morphotactically distinct stems, yielding approximately 90,000 stems based on roots and patterns. Various combinations of prefixes and suffixes, concatenated to the intersected stems, and filtered by composition, yield over 72,000,000 abstract words.

Sixty-six finite-state alternation rules map these abstract strings into fully-voweled orthographical strings, and additional orthographical relaxation rules are then applied to optionally delete short vowels and other diacritics, allowing the system to analyze unvoweled, partially voweled, and fully-voweled variant spellings of the 72,000,000 abstract words. The system is intended to serve as a pedagogical aid, a comprehension-assistance tool, and as a component in larger natural-language-processing systems.

(Darwish, 2002) [16]

Sebawai, is a shallow morphological analyzer for Arabic designed by Darwish for cross-platform. The analyzer is based on automatically derived rules and statistics (hybrid approach). The analyzer only concerned with generating the possible roots of any given Arabic word. And it has two main modules, one utilizes a list of Arabic word-root pairs, and the other accepts Arabic words as input. And it uses a collection of paired word list collected from ALPNET, Zad owned by Al-Areeb Electronic Publishers, LDC Arabic collection, and Lisan al- Arab for development and training and testing almost (579606 word). In the evaluation number of deferent size of random word are used in training and testing that yelled deferent accuracy result the best one is 96.2% using a Large training set. And 85.5% using a small training set.

(Stanford Part-Of-Speech Tagger, 2003) [7][17]

Stanford Part-Of-Speech Tagger This tagger was originally developed for English at Stanford University as Java-based open source software tagger. And then it was improved adds support for other languages together with speed and usability improvements. The tagger is based on the maximum-entropy model. The tagger was trained on the training part of the Arabic Penn Treebank (ATB); the authors claim in a 96.50% accuracy on Arabic.

(Diab and others, 2004) [18]

There work was trying to make fully automated fundamental NLP tools using make Support Vector Machine (SVM) to solve this three task tokenization at morphological level, part-of- speech (POS) tagging at lexical level and annotate base phrases (BPs) at syntactic level for MSA Arabic text. They train their SVM's system on the Arabic Penn TreeBank corpora. And SVM-TOK tokenizer achieves a score of 99.12, the SVM-POS tagger achieves an accuracy of 95.49%, and the SVM-BP chunker yields a score of 92.08.

(Habash and Rambow, 2005) [19][20]

In their approach they use ALMORGEANA morphological analyzer with Yamcha SVM-based algorithm. Using a morphological analyzer for tokenizing and morphologically tagging (including POS tagging). Train and test their approach in the Penn Arabic Tree- bank corpora (ATB1 and ATB2). By dividing them into development, training, and test corpora. Their approach has three steps. First, obtaining from the morphological analyzer a list of all possible analyses for the words of a given sentence. Second, applying classifiers for ten morphological features to the words of the text. Third, choosing among the analyses returned by the morphological analyzer by using the output of the classifiers. And then use the majority combiner to choose the analysis with the largest agreement. They use a full ATB POS tag set and a reduce one and thy yelled an accuracy score of 97.6% and an accuracy score of 98.1% in there reduced POS tag set.

(Al-Shamsi and Guessoum, 2006) [21]

A HMM POS tagger has been developed using N-gram language model. And trained and tested in a special MSA corpus build from native Arabic articles. That achieved F-measure accuracy 97%. This POS tagger contain of tokenizer, stemmer (Buckwalter's stemmer), N-gram language model. They build HMM POS tagger by constructing trigram language models and used the trigram probabilities in building the HMM model. And to solve the problem of unseen trigrams a back-off smoothing technique is used so that the model backs off to a bigram model.

(O. Smrz, 2007)[5][7][15][22][23][24]

ElixirFM and its lexicons are open-source software, that reuses and extends the Functional Morphology library for Haskell. ElixirFM uses paradigms, grammatical categories, lexemes and word classes to model inflection and derivation. And it reuses the Buckwalter lexicon and the annotations in the Prague Arabic Dependency Treebank, and linguistic model. ElixirFM reestablish the system of inflectional and inherent morphosyntactic properties (alternatively named grammatical categories or features).

The ElixirFM linguistic model and the data of the lexicon can be integrated into larger applications or used as standalone libraries and resources. ElixirFM also provides functions for exporting and pretty-printing the linguistic model into XML, L ATEX, Perl, SQL, and other custom formats.

ElixirFM provides the user with four different modes of operation (Resolve by providing tokenization and morphological analysis of the inserted text, Inflect by transforming words into the forms required by context, Derive by converts words into their counterparts of similar meaning but different grammatical category, Lookup by looking up the lexical entries of the citation form and nests of entries by the root depending on Buckwalter lexicon).

(El Hadjand others, 2009)[9]

Relying on the Arabic sentence structure a system for Arabic Part-Of-Speech Tagging is built from combining morphological analysis with Hidden Markov Model (HMM). The morphological analysis is used to reduce Arabic words lexicon by segmenting words in their prefixes, stems, and suffixes. And the HMM is used to represent the Arabic sentence structure. Using their own tagset they tag a book called "Albayan-wa-tabyin" one of ALJAHEZ's book that have been written in the of third century (Hijri). In the training and testing this system for HMM the transition' probabilities are calculated using a smoothed tri-gram, for the unknown words special processing is used. And an accuracy rate of 96% is shown from the first experiments.

(Diab, 2009) [25][26]

AMIRA 2.0 is a successor suite to the ASVMTools that process Modern Standard Arabic, used for different NLP problems such as Machine Translation, Information Retrieval, Parsing, Named Entity Recognition, Information Extraction, etc. And it includes a clitic tokenizer (TOK), part of speech tagger (POS) and base phrase chunker (BPC). The different components of the tool suite could be invoked together or directly without the need to explicitly invoke the other components. It uses Support Vector Machines in a sequence modeling framework using the YAMCHA toolkit (using character n-grams as features in sequence models). And is treat each of the component problems as a classification task based on a unified framework. The AMIRA tools are trained on Arabic Treebank.

In this version the user can choose among different of tokenization scheme, Token Count, Type of Clitic Tokenization. And the modeling of TOK is exactly the same process as that used in ASVMTools with 11 class with accuracy 99.2%. For POS Tagging optionally produces the PATB standard tag set of 25 tags (RTS) that and the extended tag set (ERTS) ERTS has 72 tags with accuracy more than 96%. 19 classification task of BPC to grouped together syntactic phrases such as NPs and VPs that gives a F1 measure of 96.33%.

(Habash and Nizar, 2009) [11] [27][28]

MADA+TOKAN is a highly configurable Toolkit for Arabic NLP applications (Tokenization, Diacritization, Morphological Disambiguation, POS Tagging, Stemming and Lemmatization). That work in of two components with the help of support utilities. MADA operates by examining a list of all possible analyses for each word, and then selecting the analysis that matches the current context best by means of support vector machine models classifying for19 distinct and n-gram statistics, weighted morphological features, all disambiguation decisions are made in one step. TOKAN takes the information provided by MADA to generate tokenized output in a wide variety of customizable formats. By that it allows users to extract and manipulate the exact information that they require. In this way they provide an excellent preprocessing tool for major NLP applications such as machine translation (MT), automatic speech recognition (ASR), named entity recognition (NER) and many others.

(Boudlal And others, 2010) [7][12][29]

Alkhalil Morpho Sys is a morphological Analyze for MSA written in Java and have a PERL version. The system has a graphic interface and it database built in xml. The system can process non vocalized texts as well as partially or totally vocalized ones. Alkhalil Morpho Sys linguistic resources are made of a set of classes, each of which

representing a category of linguistic data of the same nature and morphological features. And the tag set contain of 8 tag with sub tag. Analysis is carried out in the following steps: preprocessing, removal of diacritics; segmentation, each word is considered as (pro- clitic+stem+enclitic) and aims to identify them; analysis of the stem; result screening; display of result. The output result file is a table of 8 columns that correspond to tags and an average number of about 5 lines for a word. The table have POS-tags, prefix, suffix, pattern, stem, root and voweled word columns. And that is not good reusable for PoS.

(Attia and others, 2011) [1][30]

AraComLex is an open-source large-scale finite-state morphological processing toolkit for MSA distributed under the GPLv3 license. This morphological analyzer uses the lemma as the base form with broken plural extension. AraComLex used a corpus of 1,089,111,204 words from the Arabic Giga word corpus fourth edition and the Al-Jazeera web site in the training and testing. And the lexical contain four databases: one for nominal lemmas, one for verb lemmas, one for word patterns, and one for root-lemma lookup. The test corpus used 800,000 words, divided into 400,000 of what we term as Semi-Literary text and 400,000 for General News texts. The results for the Semi-Literary text was 85.73%, for the General News text 79.68% coverage.

(Habash and Others, 2012)[5]

Columbia Arabic Language and dIalect Morphological Analyzer (CALIMA) for morphological analyzer for Egyptian Arabic, Egyptian Arabic is a form of Dialectal Arabic (DA). It developed using hash-tables with a simple search algorithm by capturing and generalizing and extend orthographic coverage, and extending the Egyptian Colloquial Arabic Lexicon (ECAL) as a base for CALIMA. The lexicon tabular representation is compatible with the Standard Arabic Morphological Analyzer (SAMA) system for MSA. In the evaluation of the CALIMA there was a four deferent versions of the tool extended and Merged, the best one give accuracy of 92%, more than 8% using CALIMA alone.

(Hadni and others, 2013)[31]

In this work they manage to make POS Tagger for Classical Arabic language using hybrid approach in the NLTK tools. By integrating Hidden Markov Model (HMM) with Taani's rule-based tagging method and training and testing it with two corpora: The Holy Quran Corpus and Kalimat Corpus. There techniqueis by applying the Taani's rule-based tagging method in the non-vocalized word and all unanalyzed and misclassified word go to HMM tagger to be recognized. They prepared the two corpus's by reducing 33 tags to a simpler tags set of 3 tags which are: Noun (N), Verb (V) and Particle (P). The best obtained results for this method that been achieve by 90% tanning and 10% testing with Kalimat Corpus is 98% and been achieve by 90% tanning and 10% testing with Holy Quran Corpus is 97,60%.

(Khoufi and Boudokhane, 2013) [25]

A two stage corpus-based method was proposed for the annotation of MSA Arabic texts using machine learning approach adapting statistical method. The process occurs in a sequential mode until the annotation of the whole text. In the first stage thy segmented the text into sentences, then into words using punctuation marks, spaces and using a stem database to identify the prefixes, suffixes and clitics of each word. The second stage in a Morphological Analysis to assigning POS tag to the segmented word from the first stage using the N-gram model.

The proposed system is named the Arabic Morphological Annotation System (AMAS) and implemented using the Java programming language. In AMAS implementation the segmentation phase used the stem base of the morphological analyzer AraMorph1 that developed by Tim Buckwalter with some change to adapt MSA Arabic. The result is presented in the form of a structured XML file. The average measures for Precision, Recall and F-score are respectively 89.01%, 80.24% and 84.37%.

(Sawalha And others, 2013) [32][33]

The SALMA-Tools (Standard Arabic Language Morphological Analysis) is a collection of open-source standards (the SALMA Tag set), tools (the SALMA Tagger) and resources (the SALMA ABCLexicon) that widen the scope of Arabic word structure analysis - particularly morphological analysis, to process Arabic text corpora of different domains, formats and genres, of both vowelized and non-vowelized text.

The SALMA – Tagger consists of several modules which can be used independently or together to produce the analyses of the words. The SALMA – Tagger is a fine grained morphological analyzer which is mainly depends on linguistic information extracted from traditional Arabic grammar books and prior-knowledge broad- coverage lexical resources (the SALMA – ABCLexicon).

The SALMA Tag set is combination of eight existing Arabic tag sets. A 22 characters represent feature and the letter at that location represents a value or attribute of the morphological feature. The SALMA – Tag set is not tied to a specific tagging algorithm or theory, and other tag sets could be mapped onto this standard, to simplify and promote comparisons between and reuse of Arabic taggers and tagged corpora. This tag set has been validated in two ways. First, it was validated by proposing it as a standard for the Arabic language computing community. Second, an empirical approach is by showing that it can be applied to an Arabic text corpus.

(Arabic Toolkit Service (ATKS), 2013) [34][35]

Arabic Toolkit Service (ATKS) is a set of NLP components that has been developed for Arabic language in Microsoft Advanced Technology Lab in Cairo. It provides eight different NLP tool, a Colloquial to Arabic Converter, Diacritizer, Named Entity Recognizer (NER), Parser, Part of Speech Tagger, SARF (morphological analyzer), Speller, and Transliterator. That are used in most Microsoft product.

There is no published academic paper to be found to describe the ATKS tool or how it works; but Alosaimy describe SARF by acknowledging that it provided all possible analyses of a given word (affixes, stem, diacritized form and morphological features like gender), and that the TAGSET contains 109 possible complex tags. Also POS Tagger that identifies the part-of-speech of each word in a text, and the grammatical features; in addition, it resolves the nunation (the addition of nun sound that indicates noun's indefinite case). The tagger uses spelling corrector as a preprocessing step. And estimated to have more than 3000 tags in his TAGSET.

(Pasha et al, 2014) [27]

A new version of MADA, called MADAMIRA, it is Java NLP tools for MSA or EGY that combine MADA (Habash and Rambow) and AMIRA (Diab). MADAMIRA follows the same general design as MADA, with some additional components inspired by AMIRA. MADAMIRA trained on

the Penn Arabic Treebank corpus (parts 1, 2 and 3) for MSA, and the Egyptian Arabic Treebanks (parts 1 through 6) for EGY. MADAMIRA have several parts for preprocessor they use a tool to converts it to the Buckwalter representation. Then morphological Analysis using SAMA Analyzer and CALIMA Analyzer.

Then feature modeling using language models and SVM models. Then ranks analysis lists based on model predictions. Then tokenization using morphological feature. Then base phrase chunking using SVM models. Then named entity recognizer using SVM models. To evaluate MADAMIRA, a blind test data set (about 25K words for MSA and about 20K words for EGY) was run through MADAMIRA. The evaluation was conducted across several accuracies.

(Althobaiti And others, 2014)[36]

AraNLP is a free online Java-based toolkit for the processing of Arabic text. That covers various Arabic text preprocessing tools. AraNLP is an attempt to gather most of the vital Arabic text preprocessing tools into one library. The library includes a sentence detector, tokenizer, light stemmer, root stemmer, part-of-speech tagger, word segmenter, normalizer, and a punctuation and diacritic remover. It supports the most important preprocessing steps, such as diacritic and punctuation removal, tokenization, sentence segmentation, part-of- speech tagging, root stemming, light stemming, and word segmentation.

Sentence boundary detection (SBD) is the process of isolating independent sentences. A maximum entropy model for identifying sentence boundaries in raw Arabic text was build and trained on corpus collected from Arabic Wikipedia documents of various genres.

A simple tokenization (TOK) is used which only splits off punctuation and non-alphanumeric characters from words. This complex tokenization is usually called word segmentation. A MaxEnt machine learning model is builed to detects token boundaries and trained on corpus we used consists of around 52,000 tokens from the Arabic Wikipedia collection. A testing corpus with 21,000 tokens was used to evaluate the trained tokenizer, which achieved a 0.97 precision and recall score.In the stemming process they used two stemmers the first is a light stemmer like those suggested by Larkey, and the second is the root stemmer (Khoja Stemmer).

Word Segmentation & POS Tagging in AraNLP library links up to the Stanford Arabic word segmenter and POS tagger. The segmenter produces the three Penn Arabic Treebank (PATB) clitic segmentations: conjunctions, prepositions, and pronouns. The main advantage of this word segmenter is that it processes raw text quickly in comparison to other word segmenters, as its implementation is based on a sequence classifier (Conditional Random Fields). The Stanford POS tagger is based on a maximum- entropy technique.

AraNLP provides a different level of orthographic normalization that can be carried out on Arabic text to reduce noise and data sparsity. AraNLP enables the user to customize the level of normalization according to the application's need. In addition, the punctuation can easily be added or deleted from the list of punctuation marks.

(Alosaimy, 2016)[37][4]

SAWAREF is a web based multi-component toolkit that provide 8 morphological analyzers and 7 part-of-speech taggers and evaluated. And the morpological analysers included are: AlKhalil, Buckwalter, Elixir-FM, Microsoft ATKS Sarf, ALMORGEANA, AraComLex, and Xerox. And the POS taggers are: Madamira, MADA, AMIRA, Stanford POS tagger, Microsoft ATKS POS Tagger, MarMoT, CRF-based Arabic Model POS tagger using Wapiti.

The toolkit has 6 stage in general, it begins with preprocessing, tagging, parsing the result, word aligner and mapping, the tagset mapping, morphological alignment, solution alignment, ensemble and POS disambiguation. The result of all the combined component are then standardized. The standardized outputs combine different solutions, and analyze and vote for the best candidates.

Table I. Summary Timeline of Different Approaches

System name	Approaches	POS	Morphological analyzer	year	Language type	Corpora	Train	Test	Accuracy	
MORPH2[10][11] **[33]**	H	0	1	2000	MSA DA	51404 words	--	--	Recall measure: 89, 77%. Precision measure: 82, 51%.	
MORPHO3 [11][10]	H	0	1	2000	MSA CA				Coverage	Disambiguation
						Small Size	15,000 words		98%	70.21%
							50,000 words		98.73%	79.34%
						Large Size	250,000 words		98.9%	88%
							500,000 words		99.05%	94.3%
AraParse [12][13]	H	0	1	2001	MSA	AraParse 13 Mb	--	--	--	
APT [2]	H	1	0	2001	MSA	Newspaper corpus: - Al-Jazirah - Saudi. - Al-Ahram - Egyptian. - Al-Bayan - Qatari. - Al-Mishkat - Egyptian.	50,000 words from Al-Jazirah.	59,040 - Al-Jazirah 3,104 - `` Al-Ahram'' 5,811 - ``Al-Bayan'' 17,204 - Al-Mishkat.	90%	
Xerox [14][7][15]	M	1	1	Sens 2001	MSA	- Lexicons - 4930 roots. - Dictionary - 400 phonologically. - 72,000,000 abstract words.				
Darwish's Sebawai system [16]	H	0	1	2002	MSA	Collected from pair list from: ALPNET, Zad, LDC, and Lisan al-Arab.	579606 word	270000 word	96.2% Large training set. 85.5% small training set.	
Stanford POS Tagger [7][17]	S	1	0	2003	MSA	PATB	--	--	96.50%	
Diab and others, approach 2004 [18]	?	1	0	2004	MSA	PATB-ATB1	4000 sentences	400 sentences	SVM-TOK tokenizer: 99.12% SVM-POS tagger: 95.49% SVM-BP chunker: 92.08%	
Habash and	?	1	1	2005	MSA	PATB - ATB1 and ATB2,	120,000 word	12,000 word	accuracy score of 97.6%	

Rambow approach 2005 [19][20]						BAMA			accuracy score of 98.1% in there reduced POS tag set.
Al-Shamsi and Guessoum approach 2006 [21]	S	1	0	2006	MSA	Almost 10 MBs of word from native Arabic articles	9.15 MB of words	6k (944 words).	F-measure: 97%
ElixirFM [22][7][23][24]	?	0	1	2007	MSA DA	Prague Arabic Dependency Bank.			
El Hadj and others, approach 2009 [9]	S	1	1	2009	MSA	21882 words with a 3565 unique words in > 1600 sentences.	95%	5%	F-measure: 96%
AMIRA 2.0 [25][26]	?	1	1	2009	MSA DA	-	-	-	TOK 99.2% F-score measure. POS taggers: - ERTS is 96.13% - RTS 96.15%. BPC F1 measure of 96.33%.
MADA + TOKAN Toolkit **[11]** [27][28]	?	1	1	2009	MSA	PATB	PATB		Basic morphological choice and lemmatization: 96% accuracy. Predicting Full diacritization : 86% accuracy.
Alkhalil Morpho Sys [12][29][7]	?	0	1	2010	MSA				
AraComLex [1][30]	S	0	1	2011	MSA	1,089,111,204 wordsfrom the Arabic Giga word corpus fourth edition and news articles collected from the Al-Jazeera		400,000 words	85.73% Semi-Literary
								400,000 words	79.68% General News text.
CALIMA [5]	H	1	1	2012	MSA DA	ECAL - 66K entries	--	--	92%,
Hadni and others, approach 2013 [31]	H	1	0	2013	CA	- The Kalimat Corpus have a 20,291 Arabic articles - The Holy Quran corpus consists of 6236 sentences with total of 77430 words	Training%	Testing%	Holy Quran / Kalimat
							30%	70%	97% / 97,40%
							70%	30%	97,40% / 97,80%
							80%	20%	97,60% / 97,80%
							90%	10%	97,60% / 98%
AMAS [25]	S	1	1	2013	MSA	- PATB. - EASC.	Trained on ATB on 599 unvowelled texts.	Tested on 22 texts containing 10148 segmented words from EASC.	The average measures: - Precision 89.01% - Recall 80.24% - F-score 84.37%.
Microsoft ATKS POS Tagger [34][35]	?	1	1	After 2007	MSA	--	--	--	--
SALMA-Tools [32][33]	?	1	1	2013	MSA CA	1000-words from: - The Qur'an - chapter 29. - CCA.			The prediction accuracy in the text sample: - 53.50% of the Qur'an. - 71.21% of the CCA.
MADAMIRA [27]	?	1	1	2014	MSA EGY DA	- PATB (ATB1, 2 and 3) for MSA - Egyptian Arabic Treebanks (parts 1 through 6) for EGY	-	- 25K MSA words - 20K EGY words	MSA / EGY
									EVALDIAC 86.3 / 83.2
									EVALLEX 96.0 / 87.8
									EVALPOS 95.9 / 92.4
									EVALFULL 84.1 / 77.3
									EVALATBTOK
									Perfect Tokenization 98.9 / 96.6
									Correct Segmentation 99.2 / 97.6
AraNLP [36]	?	1	1	2014	MSA	- Corpus from 59 Arabic Wikipedia documents.	- SBD on 1,838 sentences. - TOK on 52,000 tokens.	- SBD on 871 sentences. - TOK on 21,000 tokens	- SBD : 0.97 precision, nearly 0.98 Recall. - TOK : 0.97 precision and recall score.
SAWAREF [37]	?	1	1	2016	MSA CA	---	---	---	---

PATB = Penn Arabic Treebank contain from ATB1, ATB2, ATB3, and ATB4.
ECAL =The Egyptian Colloquial Arabic Lexicon.
EASC = The Essex Summaries Arabic Corpus.
CALIMA = Columbia Arabic Language andDIalect Morphological Analyzer for EGY.
CCA = The Corpus of Contemporary Arabic.
SALMA-Tools = Standard Arabic Language Morphological Analysis.
Xerox = Arabic Morphological Analysis and Generation
FST = Finite-State Transducer
AMAS - Arabic Morphological Annotation System
CA: Classical Arabic–MSA: Modern Standard Arabic
Approaches: S = Statistical, M = Math, H = Hybrid,

CONCLUSION

In this paper, we presented research in a core aspect of Arabic NLP as a timeline or a list of events in chronological order, to illustrate the evolutionary development of the field. We presented a timeline of 24 different approaches and tools for Arabic Part of Speech (POS) tagging and morphological analysis. Most of the work focuses on Modern Standard Arabic (MSA). A few systems aim at Dialect Arabic (DA); the Classical Arabic (CA) gets the least attention; however, it is the core of much Modern Arabic language, and it is still widely used in religious and other contexts. We intend to explore and develop NLP modelsand tools for Classical Arabic.

REFERENCES

[1] M. Attia, P. Pecina, A. Toral, L. Tounsi, and J. Van Genabith, "A lexical database for modern standard Arabic interoperable with a finite state morphological transducer," Commun. Comput. Inf. Sci., vol. 100 CCIS, pp. 98–118, 2011.

[2] S. Khoja, "APT : Arabic Part-Of-speech Tagger," Proc. Student Work. NAACL, pp. 20--25, 2001.

[3] N. Habash, R. Roth, O. Rambow, R. Eskander, and N. Tomeh, "Morphological analysis and disambiguation for dialectal Arabic," Proc. NAACL-HLT, no. June, pp. 426–432, 2013.

[4] A. M. S. Alosaimy and E. Atwell, "Ensemble Morphosyntactic Analyser for Classical Arabic," 2nd Int. Conf. Arab. Comput. Linguist., pp. 3–9, 2016.

[5] N. Habash, R. Eskander, and A. Hawwari, "A Morphological Analyzer for Egyptian Arabic," Proc. Twelfth Meet. Spec. Interes. Gr. Comput. Morphol. Phonol. SIGMORPHON2012, pp. 1–9, 2012.

[6] D. Cutting, J. Kupiec, J. Pedersen, and P. Sibun, "A practical part-of-speech tagger," Proc. third Conf. Appl. Nat. Lang. Process. -, p. 133, 1992.

[7] H. Rabiee, "Arabic Language Analysis Toolkit," 2011.

[8] M. Sawalha and E. Atwell, "Constructing and Using Broad-coverage Lexical Resource for Enhancing Morphological Analysis of Arabic," Proc. Seventh Conf. Int. Lang. Resour. Eval. (LREC'10), Eur. Lang. Resour. Assoc., pp. 282–287, 2010.

[9] Y. El Hadj, I. Al-Sughayeir, and A. Al-Ansari, "Arabic part-of-speech tagging using the sentence structure," Proc. Second Int. Conf. Arab. Lang. Resour. Tools, vol. 0, no. 2001, pp. 241–245, 2009.

[10] Mohamed Attia Mohamed Elaraby Ahmed, "A large-scale computational processor of the arabic morphology, and applications. Thesis submitted to the of Master of science in Computer engineering," no. January, 2000.

[11] N. Kammoun, L. Belguith, and A. Hamadou, "The MORPH2 new version: A robust morphological analyzer for Arabic texts," JADT 2010 10th Int. Conf. Stat. Anal. Textual Data, pp. 1033–1044, 2010.

[12] A. Boudlal, A. Lakhouaja, A. Mazroui, A. Meziane, M. Ould Abdallahi Ould Bebah, and M. Shoul, "Alkhalil Morpho SYS1: A Morphosyntactic Analysis System for Arabic Texts," Int. Arab Conf. Inf. Technol., pp. 1–6, 2010.

[13] R. Ouersighni, "A major offshoot of the DIINAR-MBC project: AraParse, a morphosyntactic analyzer for unvowelled Arabic texts," ACL 2001 Work. Data-Driven Mach. Transl., p. 8, 2001.

[14] K. Beesley, "Finite-state morphological analysis and generation of Arabic at Xerox Research: Status and plans in 2001," ACL Work. Arab. Lang. Process. Status Perspect., pp. 1–8, 2001.

[15] M. Sawalha, "Open-source resources and standards for Arabic word structure analysis: Fine grained morphological analysis of Arabic text corpora," Univ. Leeds, 2011.

[16] K. Darwish, "Building a shallow Arabic Morphological Analyzer in one day," Proc. ACL-02 Work. Comput. approaches to Semit. Lang. -, pp. 1–8, 2002.

[17] D. Wp, T. Report, D. S. L. Final, and L. It, "Survey of POS taggers," pp. 1–12, 2013.

[18] M. Diab, K. Hacioglu, and D. Jurafsky, "Automatic Tagging of Arabic Text: From Raw Text to Base Phrase Chunks," HLT-NAACL 2004 Short Pap., pp. 149–152, 2004.

[19] N. Habash and O. Rambow, "Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop," Proc. 43rd Annu. Meet. Assoc. Comput. Linguist. - ACL '05, no. June, pp. 573–580, 2005.

[20] A. Soudi, G. Neumann, and A. van den Bosch, Arabic Computational Morphology: Knowledge-based and Empirical Methods. .

[21] F. Al-Shamsi and A. Guessoum, "A Hidden Markov Model – Based POS Tagger for Arabic," Jadt, vol. 8, 2006.

[22] O. Smrz, "ElixirFM -- Implementation of Functional Arabic Morphology," Proc. 2007 Work. Comput. Approaches to Semit. Lang. Common Issues Resour., no. June, pp. 1–8, 2007.

[23] J. Hajič, O. Smrz, P. Zemánek, J. Šnaidauf, and E. Beška, "Prague Arabic dependency treebank: Development in data and tools," Proc. NEMLAR Intern. Conf. Arab. Lang. Resour. Tools, pp. 110–117, 2004.

[24] Y. Marton, N. Habash, and O. Rambow, "Dependency Parsing of Modern Standard Arabic with Lexical and Inflectional Features," Comput. Linguist., vol. 39, no. 1, pp. 161–194, 2013.

[25] N. Khoufi and M. Boudokhane, "Statistical-based System for Morphological Annotation of Arabic Texts," Proc. Student Res. Work. Assoc. with RANLP 2013, no. September, pp. 100–106, 2013.

[26] M. Diab, "Second Generation AMIRA Tools for Arabic Processing : Fast and Robust Tokenization , POS tagging , and Base Phrase Chunking," Proc. Second Int. Conf. Arab. Lang. Resour. Tools, pp. 285–288, 2009.

[27] A. Pasha et al., "MADAMIRA : A Fast , Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic," Proc. 9th Lang. Resour. Eval. Conf., pp. 1094–1101, 2014.

[28] N. Habash, O. Rambow, and R. Roth, "MADA+TOKAN: A Toolkit for Arabic Tokenization, Diacritization, Morphological Disambiguation, POS Tagging, Stemming and Lemmatization," Proc. Second Int. Conf. Arab. Lang. Resour. Tools, no. November 2015, pp. 102–109, 2009.

[29] A. M. S. Alosaimy and E. S. Atwell, "A review of morphosyntactic analysers and tag-sets for Arabic corpus linguistics," Corpus Linguist. 2015, pp. 16–19, 2015.

[30] ANLP 2014 The EMNLP 2014 Workshop on Arabic Natural Language Processing Proceedings of the Workshop Doha , Qatar. 2014.

[31] M. Hadni, S. Alaoui Ouatik, A. Lachkar, and M. Meknassi, "Hybrid Part-Of-Speech Tagger for Non-Vocalized Arabic Text," Int. J. Nat. Lang. Comput., vol. 2, no. 6, pp. 1–15, 2013.

[32] M. Sawalha, E. Atwell, and M. A. M. Abushariah, "SALMA: Standard arabic language morphological analysis," 2013 1st Int. Conf. Commun. Signal Process. Their Appl. ICCSPA 2013, 2013.

[33] C. Brierley, M. Sawalha, B. Heselwood, and E. Atwell, "A Verified Arabic-IPA Mapping for Arabic Transcription Technology, Informed by Quranic Recitation, Traditional Arabic Linguistics, and Modern Phonetics," J. Semit. Stud., vol. 61, no. 1, pp. 157–186, 2016.

[34] A. Alosaimy and E. Atwell, "Tagging Classical Arabic Text using Available Morphological Analysers and Part of Speech Taggers," J. Lang. Technol. Comput. Linguist., vol. 32, no. 1, pp. 1–26, 2017.

[35] "Arabic Toolkit Service (ATKS)," Microsoft Advanced Technology Lab Cairo, 2013. [Online]. Available: https://www.microsoft.com/en-us/research/project/arabic-toolkit-service-atks/. [Accessed: 10-Aug-2018].
[36] M. Althobaiti, U. Kruschwitz, and M. Poesio, "AraNLP: a Java-based Library for the Processing of Arabic Text.," Proc. Ninth Int. Conf. Lang. Resour. Eval., pp. 4134–4138, 2014.
[37] D. Marneffe, "SAWAREF : Multi-component Toolkit for Arabic morphosytactic tagging," pp. 1–3, 2007.

SHORT BIODATA OF THE AUTHORS

Mohammad Elsheikh is Head of the Department of Computer Science at Shendi University, Sudan, and a researcher at the College of Computer Science and Information Technology, Sudan University of Science and Technology, Khartoum, Sudan. Mohammad specializes in research and teaching in computing, statistics and artificial intelligence.

Eric Atwell is Professor of Artificial Intelligence for Language in the Artificial Intelligence research group of the School of Computing at Leeds University, UK, and a visiting professor at the College of Computer Science and Information Technology, Sudan University of Science and Technology, Khartoum, Sudan. Eric specializes in research and teaching in data mining and text analytics, corpus linguistics, and applications in Arabic and Islamic Studies.