Deterministic Finite State Automaton of Arabic Verb System: A Morphological Study

Mohammad Mahyoob *eflu2010@gmail.com*
Department of Languages & Translation
Faculty of Science & Arts, Alula
Taibah University,
Madina, Saudi Arabia

Abstract

Finite State Morphology serves as an important tool for investigators of natural language processing. Morphological Analysis forms an essential preprocessing step in natural language processing. This paper discusses the morphological analysis and processing of verb forms in Arabic. It focuses on the inflected verb forms and discusses the perfective, imperfective and imperatives. The deterministic finite state morphological parser for the verb forms can deal with Morphological and orthographic features of Arabic and the morphological processes which are involved in Arabic verb formation and conjugation. We use this model to generate and add all the necessary information (prefix, suffix, stem, etc.) to each morpheme of the words; so we need subtags for each morpheme. Using Finite State tool to build the computational lexicon that are usually structured with a list of the stems and affixes of the language together with a representation that tells us how words can be structured together and how the network of all forms can be represented.

Keywords: Computational Morphology, Finite-State, Arabic Verb Forms, Morphological Analysis.

1. INTRODUCTION

Semitic words can be viewed as a simple mechanism consisting of two lists: a relatively short list of templates, no more than a few hundred, for forming nouns, verbs, etc. in all their inflected forms; and a much longer list of several thousand roots [1].

Arabic language belongs to Semitic group of languages. Other languages belonging to this group are: Amharic, Aramaic, Hebrew, Tigrinya and Maltese [2]. The Arabic language grammarians organized words into three main divisions. These divisions also have sub-divisions that contain every word in Arabic language. Seen in this perspective a verb form in Arabic consists of a root and a vocalic melody in addition to the agreement affixes.

As pointed in the above lines the Semitic morphology is different in many ways. A unique characteristic of this morphology is the non-concatenative merging of roots and patterns to form words or word stems [3], [4].

In view of the facts given above the Arabic morphological analysis needs to add all the necessary information (prefix, infix, suffix, etc.) to each root or stem of the words. Further, we need technical applications that analyze Arabic words and deal with internal structure of a given word [5],[6],[7].

2. RELATED WORKS

There are many morphological analyzers have been developed and built for Arabic morphology. Many techniques have been offered by the authors. The discussion here will be for the most important and in the evaluation part the results have been shown and compared with our work.

2.1 Buckwalter Arabic Morphological Analyzer

This analyzer is considered as one of the most cited work in the literature. Many tool developers make use of the data of this analyzer for developing other computational applications [8]. The problem of this system is the long output analysis.

2.2 Xerox Arabic Morphological Analysis and Generation

It is one of the good morphological analyzers for Arabic language it is a root based using FiniteStateTechnology [9]; [10]. One of the good advantages of this analyzer is the ability to cover most of the lexical features. However, it is a rule- based and English gloss is provided for each lexeme. The generation of words those are not available in the language is one of the disadvantages of this system too [11].

2.3 ElixirFM: An Arabic Morphological Analyzer by Otkar Smrz

Otakar Smrz developed an online Arabic Morphological Analyzer for Modern Standard Arabic [12]. The author made use of Buckwalter lexicon [13]. The advantage of this system is that the output of the analyzed word is processed in four different layers (Resolve, Inflect, Derive and Lookup). But the system is limited to coverage because of certain analysis.

3. VERB PARADIGMS

According to Wright [14], a great majority of the Arabic verbs is trilateral. That is to say; it contains three radical letters, though quadrilateral verbs are by no means rare. In English the infinitive form of the verb is 'to + verb' in the bare form of the verb.

But in Arabic trilateral verbs can be derived according to these scales فَعَلَ /*faʕal* ,فَعُلَ / *fuʕal*/ فَعِلَ /*faʕil*/ 'to work' . These forms constitute the first stage of verb derivation. All the derived forms can function as stems. Therefore as observed by Wright [15] , the 3rd per. sing.masc.perf. being the bare form of the verb is commonly used as paradigm. The translational equivalent of this form is the bare form of English verb which is in present tense.

The Arabic grammarians considered the verb '*fʕ l*' فعل (to work) as basic to develop a paradigm. The first radical of this trilateral verb is called by them as *fa*, the second is the *ʕ ain*, and the third is the *lam.* If we are utilizing these three base letters *fa*, *ʕ ain* and *la* we will get the noun 'action or verb'. The same thing holds true in other situation too. Thus for example, if we have three letters viz '*ra*, *sa* and *ma*' we will get the noun 'drawing' and so on. These base letters need to be woven into a pattern from the morphological system. One of these patterns is '*fʔʕ l*' which is used for the active participle.

3.1 Verb Conjunction

In order to show the verb conjugation we take a set of base letters and place them on the pattern '*fʕ l' (فعل).* These three base letters are (ر, س, م) *r, s, m. 'rsm', 'rasama*'(draw). We will study the variations of the simple past and see how the morphological inflection of the verb can change according to the verb-subject agreement.

We noticed that the root or radical form is 'r s m'. These three consonants letters are repeated in all the forms of the verb. We also find that the first and the third letters or radicals of this simple trilateral verb in the active tense is always vowelled with '*a*' *fathah*. The second letter may be vowelled by *'a*' or it can be '*u*' *Damah* or 'i' *Kassra*. The change occurs only in short vowels. However the radical letters do not change. If we compare the verb (*rsm*) with the radical *(f,* ʕ and *l*), the letter that corresponds to '*f* ' is '*r*', to 'ʕ ' corresponds 's' and to '*l*' corresponds *'m*' as shown in figure 1 below :

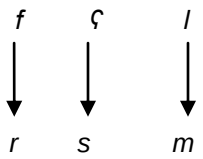

FIGURE 1*:* Radical Form of The Verb

In the imperfective active form radical form 'f ʕ l' (*faʕ al)* do not change but the vocalic melody changes. Thus we have forms '*jafʕ al', 'nafʕ al', 'ʔafʕ al'* and '*tafʕ al*' etc. We notice that the conjugation of the imperfective form is different from the base form of perfective form. There is no prefix in the perfective, only suffix is added while in the imperfective we have both the prefix and the suffix in addition to the vocalic change.

3.2 Affixation In Arabic Verbs

As we have seen above in Standard Arabic, the verb occurs in two morphological forms: perfective and imperfective. The main difference between the two is in the realization of their agreement features. In the perfective all agreement morphology is expressed by suffixes while in the imperfective, agreement features are realized by both suffixes and prefixes. The prefixes carry person features, except the first person plural, where number is also realized on the prefix; the suffixes mainly carry number features. Gender feature is also realized on the person prefix, except in the second person singular feminine, where it is realized by a suffix [16], [17].

3.2.1 Perfective Form and Affixation

The following table shows the perfective form which is realized by suffix. The verb form consists of a root and vocalic melody in addition to the agreement suffix as shown in table 1 below:

Person	Number	Gender	Affix	Verb forms
1	Singular	F/M	*-tu*	*daras-tu* '(I) studied'
2	Singular	M	*-ta*	*daras-ta*
2	Singular	F	*-ti*	*daras-ti*
3	Singular	M	*-a*	*daras-a*
3	Singular	F	*-at*	*daras-at*
2	Dual	M/F	*-tummaa*	*daras-tuma*
3	Dual	M	*-aa*	*daras-aa*
3	Dual	F	*-ataa*	*daras-ataa*
1	Plural	M/F	*-naa*	*daras-naa*
2	Plural	M	*-tum*	*daras-tum*
2	Plural	F	*-tunna*	*daras-tunna*
3	Plural	M	*-uu*	*daras-uu*
3	Plural	F	*-na*	*daras-na*

TABLE 1: Arabic Perfective Form and The Affixation.

It is claimed that there are two ways of morphological realization of the past tense:

a. "The agreement morphology suffixed to the verb realizes both tense and agreement".
b. "The vocalic melody realizes the past tense; the suffix is just a realization of the agreement morphology" [18].

Benmamoun [16], went against these two claims and he approved that the phonological realization of past tense morpheme is abstract like English simple present tense morpheme.

3.2.2 The Imperfective Form and Affixation

The imperfective in Standard Arabic occurs in different morphological forms, usually referred to as moods distinguished by their endings [19], [20].

The indicative form is represented by the sound '*u'* if the verb ends with a consonant and *ni/na* if the verb ends with a long vowel.

The subjunctive form is expressed by '*a*' if the verb ends with a consonant, but if the verb ends with a long vowel there is zero suffix. Jussive form is represented by zero morphemes. In table 2, the bare imperfective forms are shown without mood endings.

Person	Number	Gender	Affix	Verb forms
1	singular	F/M	*ʔa-*	*ʔa-ktub* I write/ am writing.
2	singular	M	*ta-*	*ta-ktub* you write/ are writing.
2	singular	F	*ta----ii*	*ta- ktub-ii* you write/ are writing.
3	singular	M	*ya-*	*ya- ktub* he writes/ is writing.
3	singular	F	*ta-*	*ta- ktub* she writes/ is writing
2	Dual	M/F	*ta----aa*	*ta- ktub-aa* you write/ are writing.
3	Dual	M/F	*ya----aa*	*ya- ktub-aa* they write/ are writing..
1	Plural	M/F	*na-*	*na-ktub* you write/ are writing.
2	Plural	M	*ta----uu*	*ta-ktub-uu* you study/ are studying
2	Plural	F	*ta----na*	*ta-ktub-na* you write/ are writing.
3	Plural	M	*ya----uu*	*ya-ktub-uu* they write/ are writing.
3	Plural	F	*ta---na*	*ta-ktub-na* they write/ are writing.

TABLE 2: Imperfective Form and Its Affixation.

3.3 Imperative

The imperative (the order or command) is formed from the imperfective form in Arabic, but there are some features for this form as stated below:

If the first radical letter is a consonant, the glottal stop is inserted at the beginning and to avoid cluster (there is no cluster in Standard Arabic word-initially) a vowel is also inserted. The insertion of this vowel is according to the vowel which follows the second radical letter of the root:

If the vowel is '*u'* the glottal stop is rendered to „*u'*.
If the vowel is '*a*' or '*I*' the glottal stop is rendered to '*I*' for example:

ta-ktubu 'you write/ are writing' changes to *ʔuktub*
taftaH „you open/ are opening' changes to *ʔiftaH*
taDrib ' you beat/ are beating' changes to *ʔiDrib*

3.4 Tense and Aspect In Standard Arabic

There is no specific indication for the tense and aspect in Arabic verb forms (perfective and imperfective) [21].

Arabic does not grammaticalize the perfective /imperfective distinction, nor does it have any particular progressive morphology [22]. As Fassi Fehri pointed out that, we will check these two examples:

a- *katab-a* wrote-3.s.m
b- ya-ktub-u write-3.s.m.Indic

In (a), we noticed that the verb carries lexical meaning of the verb, the past tense and the active voice. In (b), the verb form indicates the imperfective; the suffix indicates the indicative mood and the agreement. In these two examples we noticed that the tense morpheme in both cases is abstract. Fehri shows the relation between the agreement and affixation. Two kinds of contrasts contribute to the identification of temporal morphemes; on the one hand, we have the internal vocalic pattern, on the other, the position of the agreement morpheme. With the past forms, the agreement (with the subject) is exclusively by suffixes. With non-past forms, the agreement is both by prefixes and suffixes.

The past tense morpheme in Arabic is not realized by the overt affixes of the perfective form that seems to carry agreement only. The vocalic melody of the verb does not carry the past tense as well. It seems to be an abstract morpheme located in tense which can be hosted by negation or by the verb [23].

4. DETERMINISTIC FINITE STATE AUTOMATON (FSA)

Deterministic Finite State Automaton is a finite state machine that accepts/rejects finite strings of symbols and only produces a unique computation of the automaton for each input string. 'Deterministic' refers to the uniqueness of the computation. The behaviour of the deterministic finite state automaton during the recognition is fully determined by the state it is in and the symbol it is looking at. For example, the figure 2 illustrates a deterministic finite automaton using a state diagram. There are three states: S0, S1 and S2 which are called nodes. The automaton takes a finite sequence of 0s and 1s as input. For each state, there is a transition arrow leading to a next state for both 0 and 1. A DFA jumps deterministically from a state to another by following the transition arrow. For example, if the automaton is currently in state S0 and current input symbol is 1 then it deterministically jumps to state S1. A DFA has a start state (denoted graphically by an arrow coming in from nowhere) where computations begin, and a set of accept states (denoted graphically by a double circle) which helps define when a computation is successful1.

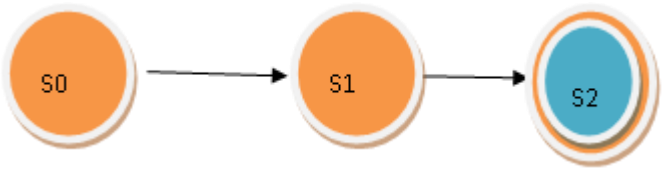

FIGURE 2: An Example of A Deterministic Finite State Automaton.

A deterministic finite automaton is a 5-tuple, $(Q, \sum, , q_0, F)$, consisting of

- (Q) a finite set of states
- (∑) a finite set of input symbols called the alphabet
- δ a transition function ($\delta : Q \times \Sigma \rightarrow Q$)
- q_0 a start state ($q_0 \in Q$)
- F a set of accept states ($F \subseteq Q$)

The machine starts in the start state q_0 or s_0, the machine will transit from state to state with the data according to the transition function δ. Finally, the machine accepts data if the last input of this data causes the machine to halt in one of the accepting states. Otherwise, it is said that the automaton rejects the string [24], [25].

4.1 Morphological Parser

To Build a Morphological Parser, we need at least the following:

- Lexicon (the list of stem and affixes together with basic information).
  This basic information is about the word stem. Lexicon is a repository for words.
- Morphotactics refers to the model of morpheme ordering. This model explains which classes of morphemes are there inside the word. In other words which morphemes precede and which follow.

There are many ways to model morphotactics . Finite State Automaton is one of these models which is discussed in this paper.

- Graphotactics (spelling rules).these rules include the deletion, the addition or transformation processes.

4.1.1 Developing Finite State Lexicon

A lexicon is a repository for words. The simplest lexicon would consist of an explicit list of every word of the language; by every word we mean every word, including abbreviations and proper nouns. It is impossible to list all the words in the language, computational lexicons are usually structured with a list of the stems and affixes of the language together with a representation of the mophotactics that tells us how they can fit together. There are many ways to model morphotactics; one of the most common is the finite state automaton, [26],[27],[28],[29], [30].

The following table is a representation of the inflection system and affixation summary in Arabic Verb Forms**.**

Aspect	**Tense**			**Root**	**Agreement**
IMPERFECTIVE	Future+ present	*Sa-* 0	*ta-* *ʔa-* *ya-* *na-*	*Ktb* *rsm* *nsr*	*-ii/aa/uu/na/0* *0* *-uu/aa/0* *0*
	Imperative		ʔu- ʔi-	*Ktb*	*-ii/aa/na/uu/0*
PERFECTIVE	Zero morpheme			*Katab* *rasam* *nasar* *nasar* *nassar* *kasar* *kassar*	- *t+u/a/i/ummaa/u m/unna* *a+t/a/taa* *-na+a* *Uu*

TABLE 3: The Affixation Summary In Arabic Verb Forms.

4.2 Experiment

In the following table we will see an example of building lexicon for some verbs in perfective forms.

In this lexicon we have a list of three verbs: *katab (*write*), rasam (*draw*) and ishtra (*buy*)* and a list of all possible affixation for the perfective forms which can be shown in subject- verb agreement for gender, person and number as shown in table 4.

Multichar_Symbols +SG +DL +PL +N +V +1P +2P +3P +PERF +NOM +ACC +MA +FE [+SG stands for singular, +DL stands for dual +PL stands for plural, +N stands for Noun, +V stands for Verb, +1P stands for 1st person singular|dual|plural, +2P stands for 2nd person singular|dual|plural, +3P stands for 3rd person singular|dual|plural, +PERF stands for perfective, +NOM stands for nominative case, +ACC stands for accusative case, +MA stands for masculine, +FE stands for feminine]
LEXICON Root Verbs;
LEXICON Verbs
katab Vend;
rasam Vend;
ishtara Vend;
LEXICON Vend
+V:0 #;
+V+1P+SG+PERF:tu #;
+V+2P+SG+MA+PERF:ta #;
+V+2P+SG+FE+PERF:ti #;
+V+3P+SG+MA+PERF:a #;
+V+3P+SG+FE+PERF:at #;
+V+2P+DL+MA+PERF:tuma #;
+V+2P+DL+FE+PERF:tuma #;
+V+3P+DL+MA+PERF:aa #;
+V+3P+DL+FE+PERF:ata #;
+V+1P+PL+PERF:na #;
+V+2P+PL+MA+PERF:tum #;
+V+2P+PL+FE+PERF:tunna #;
+V+3P+PL+MA+PERF:u #;
+V+3P+pL+FE+PERF:nna #;

TABLE 4: An example of Lexicon for some verbs in perfective forms

The first column contains the bare form of the verb and it's tense. The second column contains the stem of each word and its entire morphological features. These features give additional information about each word stem. The feature +V indicates that the word is verb; +SG indicates that the word is singular;+DL means that the word is dual; +PL means that the word is in the plural form; +MAS means that the word is masculine; +FEM means the word is feminine in gender.

According to table4 we believe that the task or the main goal of morphological analysis is to list all possible analysis of the words.

The following figure shows the model of processing verb morphology in Arabic language.

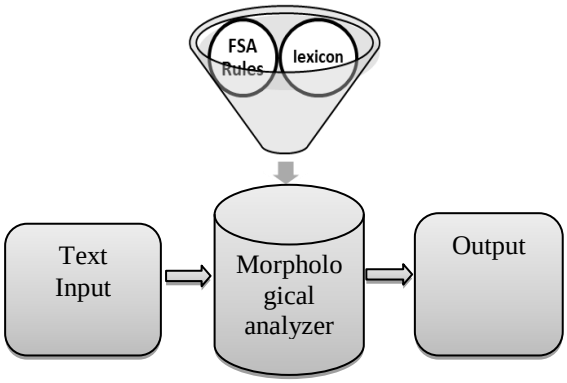

FIGURE 3: Model of Our Study.

4.3 Results and Analysis

The finite state automaton for Arabic verb forms given in Table 3 can be represented diagrammatically as follows:

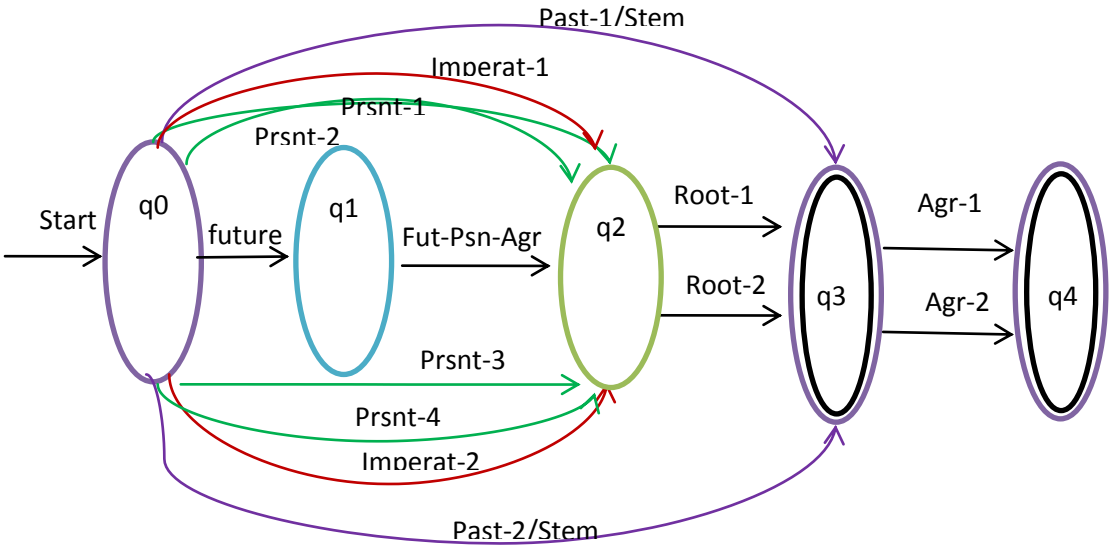

FIGURE 4: A Deterministic Finite State Automation for the verb forms in Arabic

In figure 4: we have seen the following:
The number of the states are: 5
The number of the transitions (arcs) are: 14

q0 is the initial state
q3 and **q4** are the final states
Σ: {***sa*** (future), ʔ**u**(Imperative-1), ʔ **I (** Imperative-2), ***ta*** (Prsnt-1), ʔ***a (*** Prsnt-2), ***ya*** (Prsnt-3), ***ta***/ ***ʔ a/ya/na***(fut-psn-agr), ***ksr*** (Past-1), ***kssr*** (past-2), ***ktb*** (root-1), ***rsm*** (root-2), ***-ii/aa/uu/na*** (Agr-1), -***t+i/u/a/ummaa/uu/unna/, a+t/a/taa, na+a,uu*** (agr-2)}

4.4 The Symbols

The following are the symbols which are used in the Finite State Automation (Figure 2):

1- **Past-1: {*ksr*}** this symbol refers to the past tense form which can be the first bare form of the verb conjugation in Standard Arabic, no prefix precedes past tense*.
2- **Past-2: {*kssr*}** this symbol refers to the past tense form which can be the second bare form of the verb conjugation in Standard Arabic.
3- **Future**: **{*sa*}** is applicable for future tense marker. It has to be followed by the prefix of present markers.
4- **Imperative-1**: the imperative marker which can be added to the root directly, it has two morphemes:
   The morpheme {ʔ**u**} is applicable for the imperative when the second radical consonant is followed by *'u'* as shown in this example:
   ta-ktubu 'you write/ are writing' changes to ʔ*uktub*
5- **Imperative -2 {ʔ i}** is applicable when the second radical letter of the root is either '*i*' or '***a***' as in the following examples:
   *taft**a**H* 'you open/ are opening' changes to ʔ **i***ft**a**H*
   *taDr**i**b* ' you beat/ are beating' changes to ʔ ***i****Dr**i**b*
6- **Fut-psn-Agr (future-person-agreement) { *ta*/ *ʔ a/ya/na}** the future tense morpheme has to be followed by the agreement of persons which are the same as in present tense morphemes.

7- **Prsnt-1**: ***{ta}*** is applicable for present tense form with the following person, gender and number: second singular feminine, second dual masculine/feminine, second plural masculine/feminine, third singular female, and third plural feminine.
8- **Prsnt**-2: ***{ʔa}*** is applicable for first singular feminine/masculine.
9- **Prsnt-3: {ya}** is applicable for third singular masculine, third dual masculine / feminine and third plural masculine.
10- **Prsnt-4: {na}** is applicable for first plural masculine/feminine.
11- **Root-1:** ***{k t b}***
12- **Root-**2: ***{r s m}***
13- **Agr-**1: ***{-ii/aa/uu/na}*** is applicable for present, future and imperative.
14- **Agr-**2: ***{-t+u/i/ummaa/um/unna, -a+t/a/taa, -na+a, -uu***} is applicable for perfective (past tense).

4.5 Transition Function Matrix

Transition function matrix between the states indicates how the transition moved from one state to another carrying some data.

In the following table we will show the number of states and how the transition function matrix moves from one state to another according to figure 4 above.

From	To	Output
0	1	*sa* -(future)
0	2	*ta*- (Prsnt-1)
0	2	*ʔa*- (Prsnt-2)
0	2	*ta*- (prsnt-3)
0	2	*na*- (prsnt-4)
0	2	*ʔ u- (imperative-1)*
0	2	*ʔ i -(imperative-2)*
0	3	*ksr* (past-1)
0	3	*kssr* (past-2)
1	2	*Fut-Pre-Agr*
2	3	*Roots*
3	4	*-ii/aa/uu/na* (Agreement-1)
3	4	*-t+u/i/ummaa/um/unna, -a+t/a/taa, -na+a, -uu* (Agreement-2)

TABLE 5: The Transition Table.

3.7 Example

In Figure (4) we have shown diagrammatically how the finite state machine works. The transition of the three verbs; *katab (*wrote*), rasam (*drew*) and saafar (*travelled) is illustrated in the following figure and the output is shown in table 6 below.

Sigma= {l**, a, b, h, I, k, m, n, p, r, s, t, u, + +DL +FE +MA +PERF +PL +SG +V +1P +2P +3Pa}**

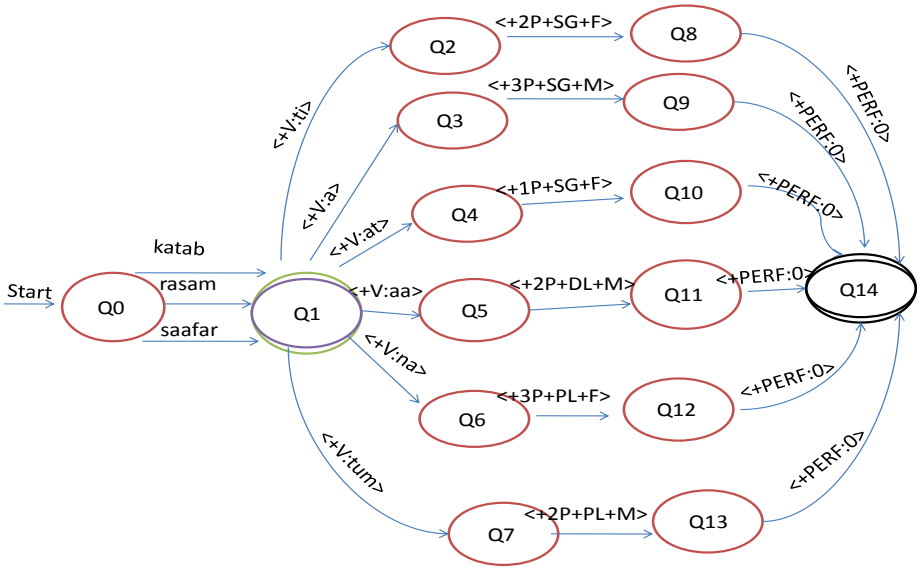

FIGURE 5: States Transition Diagram for *katab, rasam, and saafar.*

The above diagram shows the automatic network representation and how the Finite State machine is working to produce the inflected forms of the verbs.

Table 6 below shows the output of our morphological analysis using FSA, by testing our developed program the outcomes are possibly all the forms of the verbs according the tense either perfective or imperfective.

Input	Morphological Parsed Output
r a s a m (drew) perfective forms	*r a s a m +V* *r a s a m 0* *r a s a m +V +1P +SG +PERF* *r a s a m t u 0 0* *r a s a m +V +2P +SG +MA +PERF* *r a s a m t a 0 0 0* *r a s a m +V +2P +SG +FE +PERF* *r a s a m t i 0 0 0* *r a s a m +V +2P +DL +MA +PERF* *r a s a m t u m a 0* *r a s a m +V +2P +DL +FE +PERF* *r a s a m t u m a 0* *r a s a m +V +2P +PL +MA +PERF* *r a s a m t u m 0 0* *r a s a m +V +2P +PL +FE +PERF* *r a s a m t u n n a* *r a s a m +V +3P +SG +MA +PERF* *r a s a m a 0 0 0 0* *r a s a m +V +3P +SG +FE +PERF*

	r a s a m a t 0 0 0 *r a s a m +V +3P +DL +FE +PERF* *r a s a m a t a 0 0* *r a s a m +V +3P +DL +MA +PERF* *r a s a m a a 0 0 0* *r a s a m +V +1P +PL +PERF* *r a s a m n a 0 0* *r a s a m +V +3P + p L +FE +PERF* *r a s a m n n a 0 0 0 0* *r a s a m +V +3P +PL +MA +PERF* *r a s a m u 0 0 0 0*

TABLE 6: Output of The Perfective Forms of The Verb *drew 'rasam'.*

The goal is to process an input form, from those in the first column and produce output forms, like those in the second column as shown in table 6.

4. EVALUATION

The following table brings the obtained results of the tested data in our system.

Data	Correct output	Generated forms	F-Score
1500	96.00	50.00	65.75
750	97.00	50.00	74.08

TABLE 7: Obtained Results.

The evaluation of our work is compared with other previous morphological analyzers; Tim Buckwalter Morphological Analyzer, Tri-literal Root algorithm, Khoja Stemmer, Xerox Morphological Analyzer and ElixirFM. We performed the experimental analysis to show that the developed program outcomes are all possible forms of the verbs according the tense; either perfective or imperfective. The results generated by proposed methodology are sufficient and concrete with high accuracy of 96.00%. Our system brings the best results compared to previous systems. The advantages and disadvantages of some previous systems are discussed in the related work above. The following table shows the accuracy percentage of our system and other morphological analyzers:

Morphological Analyzers	Buckwalter morph. Analyzer	Tri-literal Root algorithm	Khoja stemmer	Xerox Morpho. Analyzer	ElixirFMS	Our system
Accuracy	33.91%	65.00%	71.25%	88.91%	89.58%	96.00%

TABLE 8: The Evaluation Process Results.

5. CONCLUSION AND FUTURE WORK

The present study, however, made several noteworthy contribution to the field of Arabic computational morphology by presenting the language analysis in a new and easy methods, by using FSA tool, and testing the obtained output.

This paper discussed how to build Finite-state machines based on the linguistic principles for the verb system of Arabic language.

It also describes the morphological analysis and processing of verb forms in Arabic using finite state machine It focuses on the inflected verb forms. It shows the methods of analyzing Arabic verbs with the morphological and orthographic features of Arabic and the morphological processes which are involved in Arabic verb formation and conjugation. The Morphological analyzer adds all the linguistic information to each morpheme of a word. We use the Finite State tool to build the computational lexicon that is usually structured with a list of the stems and affixes of the language together with a representation that tells us how they can be combined together.

This paper is a representation of Arabic Verb Forms by developing a morphological analyzer for these forms. Future plan is to cover all Arabic categories and forms .i.e. developing a morphological analyzer for Nouns forms, Pronouns forms and other Particles. This work will be extended to develop a POS Tagger and Parser for Arabic Language categories.

6. REFERENCES

[1] Shimron, J. *Language Processing and Aquisition in Language of Semitic, Root-based, Morphology.* Amsterdam: John Benjammins Publishing Company, 2002.

[2] Haywood, J.A. and Nahmad, H.M. A New Arabic Grammar of the written Language. London: Lund Humphries, 1965.

[3] Farghaly, A., and K. Shaalan. "Arabic Natural Language Processing: Challenges and Solutions", ACM Transactions on Asian Language Information Processing, 2009.

[4] Greenberg,J.1950.'The patterning of root morphemes in Semitic'.Word 6:162-81.

[5] Forsberg M. and Ranta A. Functional Morphology ICFP'04, Proceedings of the Ninth ACM SIGPLAN International Conference of Functional Programming, September 19-21, Snowbird, Utah, 2004.

[6] Atwell E., Al-Sulaiti L., Al-Osaimi S., Abu Shawar B.“A Review of Arabic Corpus Analysis Tools”, JEP-TALN 04, Arabic Language Processing, Fès, 19-22 April, 2004.

[7] Bender, M. L. Amharic Verb Morphology. East Lansing, Michigan: Michigan State University, 1978.

[8] Bat-El, O. “Semitic verb structure within a universal perspective”. Amesterdam: Language Aquisition and Language Disorder:28, 2002.

[9] Beesley K.R. “Arabic Finite-State Morphological Analysis and Generation”, Proceedings the 16th conference on Computational linguistics, Vol 1. Copenhagen, Denmark: Association for Computational Linguistics, 1996, pp 89-94.

[10] Aronoff, M. Morphology by Itself: Stem and Inflectional Classes. Cambridge: The MIT Press, 1994.

[11] Beesley KR. Finite-State Non-Concatenative Morphotactics, SIGPHON-2000, Proceedings of the Fifth Workshop of the ACL Special Interest Group in Computational Phonology, Luxembourg, August 6, 2000, p. 1-12.

[12] Darwish K. Building a Shallow Morphological Analyzer in One Day, Proceedings of the workshop on Computational Approaches to Semitic Languages in the 40th Annual Meeting of the Association for Computational Linguistics (ACL-02). Philadelphia, PA, USA, 2002.

[13] Buckwalter T. Buckwalter Arabic Morphological Analyzer Version 1.0. Linguistic Data Consortium, University of Pennsylvania, LDC Catalog No.: LDC2002L49, 2002.

[14] W. Wright, L. A Grammar of the Arabic Language. New Delhi: Munshiram Manoharlal Publishers Pvt. Ltd, 2004.

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

[16] Bnmamoun, E. The Feature Structure of Functional Categories: A Comparative Study of Arabic Dailects. Oxford: Oxford University Press, 2000.

[17] Bnmamoun, Elabbas. “The role of the imperfective template in Arabic morphology”. Language Acquisition and Language Disorders, 2003, 28: 99-114

[18] McCarthy, J. “A Prosodic theory of nonconcatenative Morphology” Linguistic Inquiry, 1981,12 (3): 373-418.

[19] Benmamoun, E. The Feature Structure of Functional Categories: A Comparative Study of Arabic Dailects. Oxford: Oxford University Press, 2000.

[20] Hassan, A. Al-NaHw Al-Wafii, . vol. 4. Cairo: Daar Al-Maarif, 1973.

[21] Fehri, F. Issues in the Sructure of Arabic Clauses and Words. Dordrecht: Kluwer, 1993.

[22] Fehri, F. Issues in the Sructure of Arabic Clauses and Words. Dordrecht: Kluwer, 1993.

[23] Beesley, K. R. and L. Karttunen, Finite State Morphology. Stanford, Calif., Csli, 2003.

[24] Jurafsky, D. and J.H.Martin. Speech and Language Processing, Prentice-Hall, New, 2000. Jersy.

[25] McCarthy, J. 1979. Formal Problems in Semitic Phonology and Morphology . Ph.D. dissertation. Camberidge:MIT.

[26] Nizar Y. Habash. 2010. Introduction to Arabic Natural Language Processing. Morgan & Claypool

[27] Roy Bar-Haim, K. S. 2006. Part-of-Speech Tagging of Modern Hebrew Text. Cambridge: Cambridge University Press.

[28] Sawalha, M., & Atwell, E. (2008). Comparative evaluation of Arabic language morphological Analysers and stemmers. Coling 2008: Companion volume: Posters, 107-110.

[29] Sproat, B. R. 2007. Computational Approaches to Morphology and Syntax. Oxford: Oxford University Press.

[30] Wikipedia contributors, 'Deterministic finite automaton', Wikipedia, The Free Encyclopedia, , 07:24 UTC: https://en.wikipedia.org/w/index.php?title=Deterministic_finite_automaton&oldid= 836856582> 17 April 2018 [4 May 2018].