

Arabic Morphology

Mohamed Atia Mohamed

Mehad Nasser

In Arabic a word can mean a sentence thanks to its compound structure, which is an agglutination of elements of grammar; the following representation outlines a possible structure of a word. Note that the reading and writing of a word is from right to left

Proclitic + Prefix + BODY + Suffix + Enclitics

Cliticization Morphology:

- **Proclitics** are prepositions and conjunctions:

Clitics	Class	PATB POS Tag	Function	English
+أ Ā+	QST	INTERROG_PART	interrogative همزة الإستفهام	yes/no question
+و wa+	CNJ	CONJ	coordination واو العطف	and
			connection واو الربط	and
		SUB_CONJ	circumstantial واو الحال	while
	PRT	PREP	oath واو القسم	by
			accompaniment واو المعية	with
+ف fa+	CNJ	CONJ	conjunction فاء العطف	and, so
		CONNEC_PART	connection فاء الربط	and, so
		RC_PART	response conditional فاء الجزاء	so, then
		SUB_CONJ	subordinating conjunction فاء السببية	so that
+ب bi+	PRT	PREP	preposition حرف جر	by, with, in
+ك ka+	PRT	PREP	preposition حرف جر	such as, like
+ل li+	PRT	PREP	preposition حرف جر	to, for
+لا la+	PRT	EMPHATIC_PART	emphasis لام التوكيد	will certainly
		RC_PART	response conditional اسمها	so, then
+س sa+	PRT	FUT_PART	future particle سين المستقبل	will
+ال Al+	DET	DET	definite article ال التعريف	the

And proclitics we used are:

ال	وَ	فَ	وَال	فَال	أَ	أَفَ	أَوْ	سَ	فُسَ
وَسَ		بَال			فِبَال	وِبَال	لَ	فَل	وَل
كَ	وَكَ	فَكَ	كَال	وَكَال	فَكَال	أَسَ	أُفَسَ	أُوسَ	أُبَ
أَكْ	أَلْ	أَبَال	أَكَال	أَلْ	أَفْلِل	أُولِل	لِلْ	فَلِلْ	وَلِلْ
أَقْبَ	أَوْبَ	أَفَكَ	أَوَكْ	أَفْلَ	أُولَ	أُفْبَال	أُوبَال	أُفْكَال	أُوكَال
بَ	فَبَ	وَبَ		لَ	فَلَ				

- **Prefixes** are particles placed at the beginning of verbs in the imperfective (the present continuous (مضارع) and the imperative (أمر)):

أَ	أُ	يَ	يُ	تَ	تُ	نَ	نُ	إِ	أَ	أُ
----	----	----	----	----	----	----	----	----	----	----

- **Suffixes:**

ـَ	ـُ	ـِ	أَ	أُ	أِ	ةَ	ةُ	ةِ	ةُ
ةَ	ةُ	ةِ	اتُ	اتُ	اتُ	اتُ	تُ	تُ	تُ
تَ	انَ	تَا	تَانِ	تَيْنِ	تَيْنِ	يَ	يَنِ	يَنِ	وُ
واُ	ونَ	يَ	يَنِ	ثَمَا	ثَمَ	ثَنَ	نَ	نَا	نَ

- **Enclitics:**

		Number		
Person	Gender	Singular	Dual	Plural
1st		1,3 يَ + iy 2 نِي + niy	نا + nA	
2nd	Masc	كَ + ka	كما + kumA	كُم + kum
	Fem	كِ + ki		كُنْ + kun~a
3rd	Masc	هُ + hu	هُمَا + humA	هُم + hum
	Fem	هَ + hA		هُنْ + hun~a

Used:

نِي	ي	نَا	كَ	كَ	كَمَا	كَمْ	كَنْ	ه
ه	هَا	ه	هَمَا	هَمَا	هَم	هَم	هَنْ	هِنْ

- Prefixes and suffixes express grammatical features and indicate the functions: event name, mode of the verb and the modalities (number, gender, person ...).
- Enclitics can be used to extract PERSON information.

Base forms, Inflectional Morphology:

• VERB morphology

There are 15 well known forms for verbs inflectional morphology :

Form	PV-Pattern	IV-Pattern	Meaning	Example	Gloss
I-V _p V _i	1a2V _p 3 (1u2i3)	a12V _i 3 (u12a3)	Basic sense of root	-	-
I-aa	1a2a3	a12a3	-	<i>fataH, y+aftaH</i>	open
I-au	1a2a3	a12u3	-	<i>katab, y+aktub</i>	write
I-ai	1a2a3	a12i3	-	<i>jalas, y+ajlis</i>	sit
I-ia	1a2i3	a12a3	-	<i>yaDib, y+aγDab</i>	be angry
I-ii	1a2i3	a12i3	-	<i>Hasib, y+aHsib</i>	consider
I-uu	1a2u3	a12u3	-	<i>Hasun, y+aHsun</i>	be beautiful
II	1a22a3 (1u22i3)	u1a22i3 (u1a22a3)	Intensification, causation	<i>kat~ab, y+ukat~ib</i>	dictate
III	1A2a3 (1uw2i3)	u1A2i3 (u1A2a3)	Interaction	<i>kAtab, y+ukAtib</i>	correspond with
IV	'a12a3 (u12i3)	u12i3 (u12a3)	Causation	<i>Aajlas, y+ujlis</i>	seat
V	ta1a22a3 (tu1u22i3)	ata1a22a3 (uta1a22a3)	Reflexive of Form II	<i>taEal~am, y+ataEal~am</i>	learn
VI	ta1A2a3 (tu1uw2i3)	ata1A2a3 (uta1A2a3)	Reflexive of Form III	<i>takAtab, y+atakAtab</i>	correspond
VII	in1a2a3 (in1u2i3)	an1a2i3 (un1a2a3)	Passive of Form I	<i>Ainkatab, y+ankatib</i>	subscribe
VIII	i1ta2a3 (i1tu2i3)	a1ta2i3 (u1ta2a3)	Acquiescence, exaggeration	<i>Aiktatab, y+aktatib</i>	register
IX	i12a3a3 (i12u3i3)	a12a3i3 (u12a3a3)	Transformation	<i>AiHmar~, y+aHmar~</i>	turn red, blush
X	ista12a3 (istu12i3)	asta12i3 (usta12a3)	Requirement	<i>Aistaktab, y+astaktib</i>	make write
QI	1a23a4 (1u23i4)	u1a23i4 (u1a23a4)	Basic sense of root	<i>zaxraf, y+uzaxrif</i>	ornament
QII	ta1a23a4 (tu1u23i4)	a1a23a4 (uta1a23a4)	Reflexive or unaccusative of QI	<i>tazaxraf, y+atazaxraf</i>	be ornamented

- **Particles:**

- Supplementation (حروف العطف) : و, ثم, etc.
- Interrogative (حروف الإستفهام) : هل, ماذا, متى : etc
- Preposition say (حروف الجر) : في, ب, etc
- Unaccomplished 'mansoub' (حروف النصب) : أن, إن, لن : etc..
- Determination (التعريف) : ال

- **Nominals:**

Source verb	Verb derived noun	Example	Meaning
عَلِمَ (alima) means "he knew"	Gerund	الْعِلْمُ (al-'Ilmu)	Knowing
	Active participle	عَالِمٌ ('aalimun)	One who knows
	Hyperbolic participle	عَالِمَةٌ ('allaamatun)	One who knows a lot
	Passive participle	مَعْلُومٌ (ma'aluumun)	That which is known
	Resembling participle	عَالِمٌ ('aliimun)	One who knows intrinsically
	Utilitarian noun	مِيعَامٌ (mi'laamun)	Through which we know
	Locative noun	مَعْلَمٌ (ma'limun)	Where/when we know
	Comparative and Superlative	أَعْلَمُ (a'lamu)	One who knows the most

IMPLEMENTATION

- **Augmented Transition Network:**

For each ATN there is an initial state, final states, and transitions (arcs) defined:

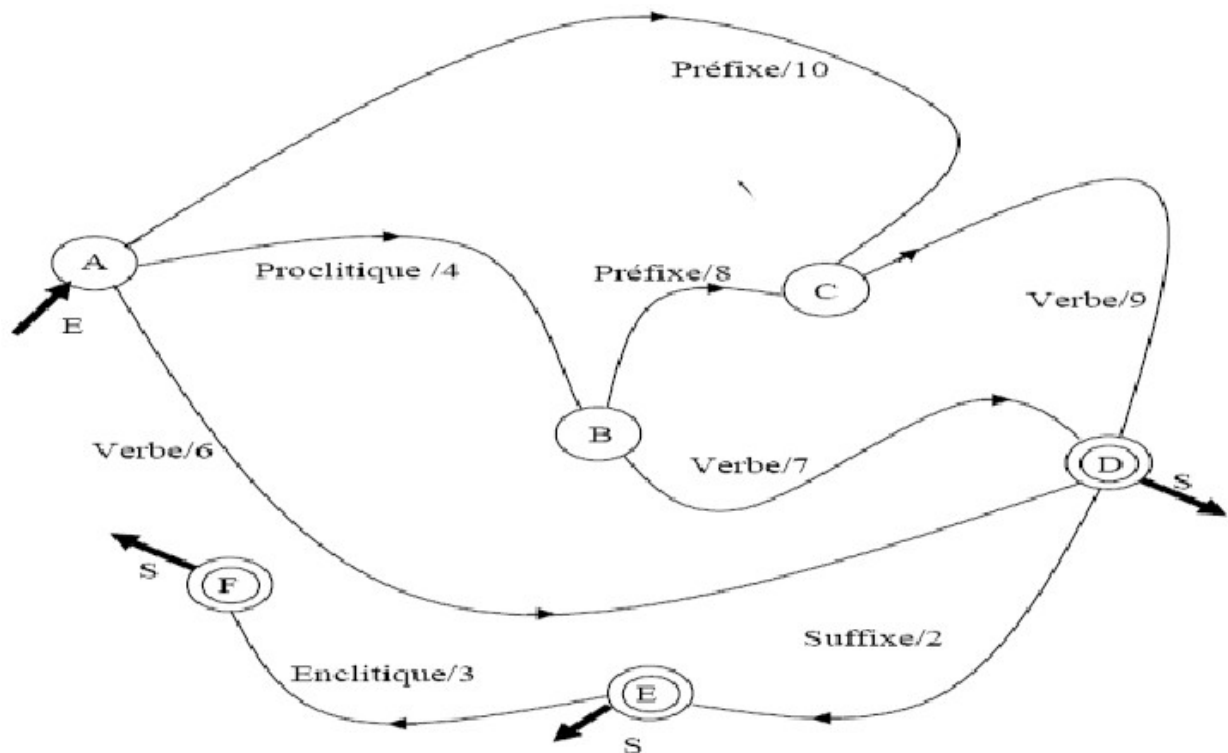
```
%word-network
initial(w, 1).
final(w, 2).
final(w, 3).
final(w, 4).
arc(w,
    1, part,
    X, Parse, Parsel):-
    Parse = [part | Parsel],
    X= 2.
arc(w,
    1, name,
    X, Parse, Parsel):-
    Parse = [name |
    Parsel],
    X= 3.
arc(w,
    1, verb,
    X, Parse, Parsel):-
    Parse = [verb | Parsel],
```

X= 4.

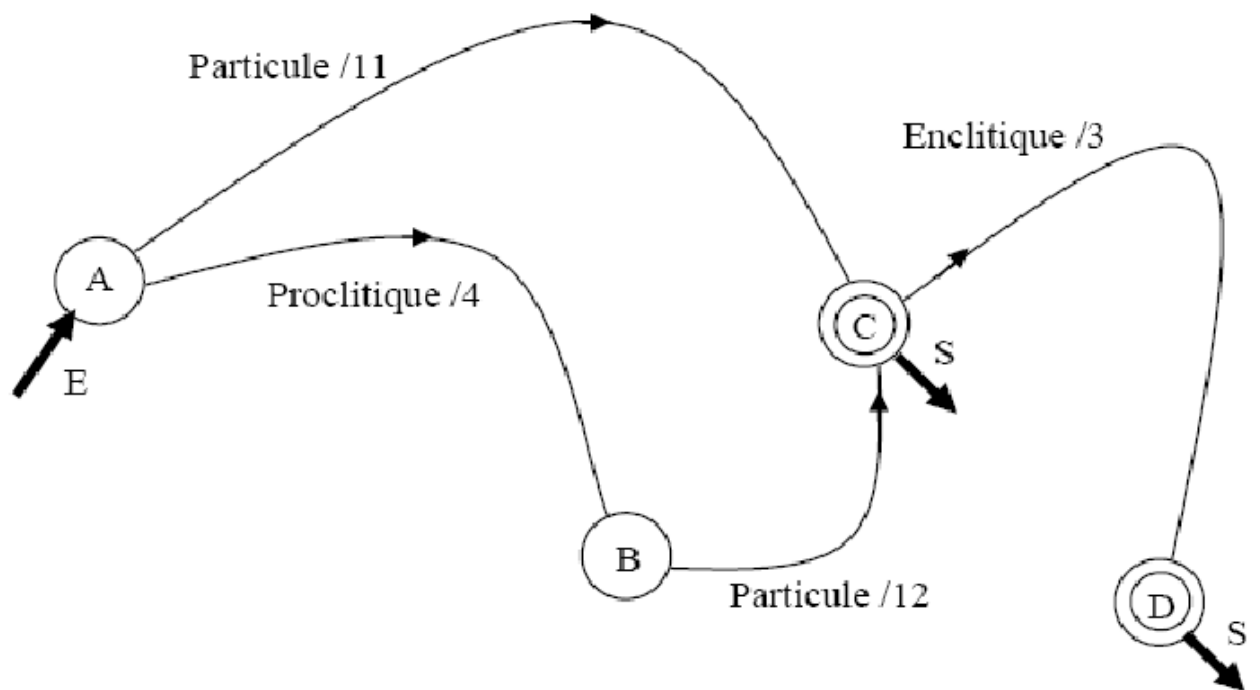
- With each network “w” any predicate has to have an input identifying it's network so it's initial(w, 1), which identifies state 1 as initial state for network w.
- Arc(network, from, term, to, outputParse, lastParse), we preferred this to be a predicate not fact form, so we can do various checks within the predicate and change the ATN registers as we need.
- Parse = [name | Parsel]: this is the output of the ATN, and can arc being a predicate makes it easy to change its format, we can easily change the output to a one collective fact instead of a list: verb(asl1(ف), asl2(ع), asl3(ج)), but found the list easier to manage and interoperate with.
- The ATN allows jump arcs which was very useful in building very compact networks:

```
arc(pref,  
2, jump,  
X, Parse, Parsel):-  
Parse = Parsel,  
X= 3.
```

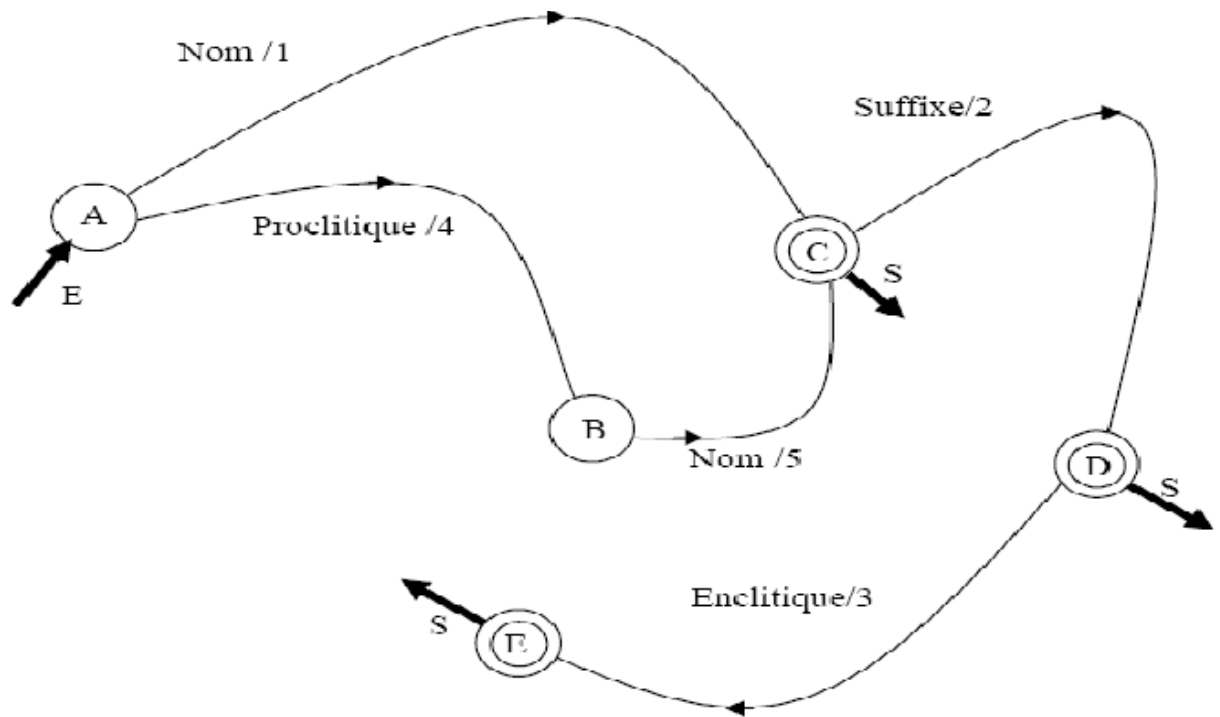
- Verb ATN:



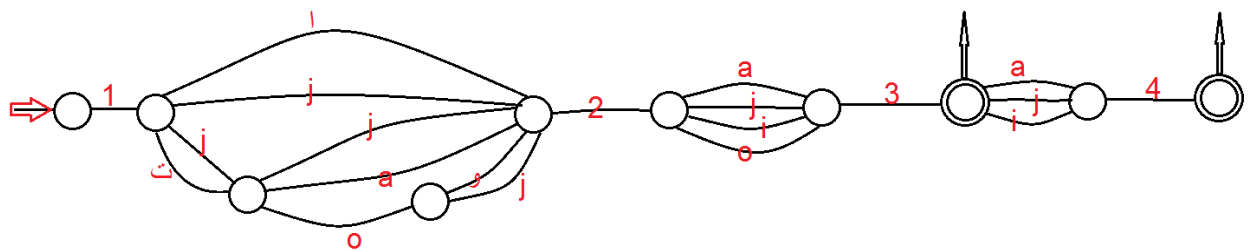
- Particle ATN:



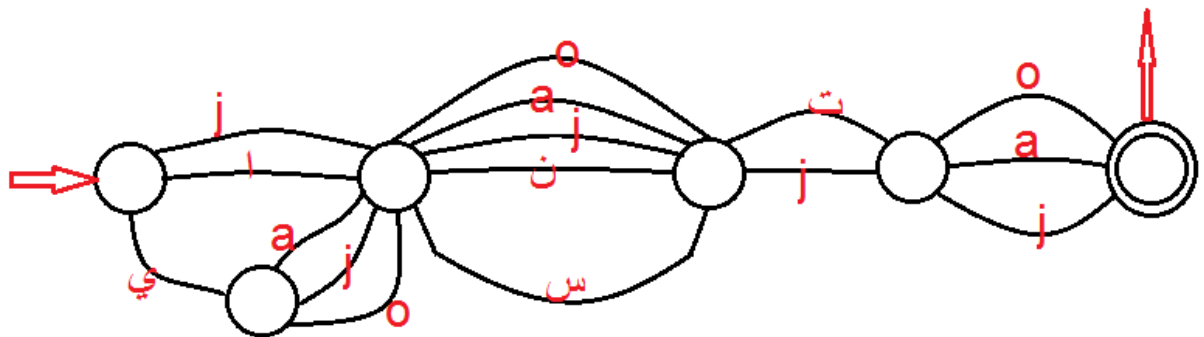
- Name ATN



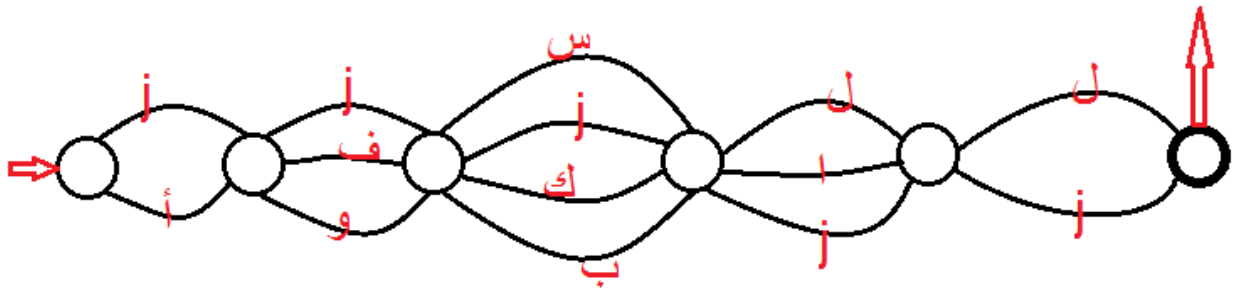
- ASL (Verbe):



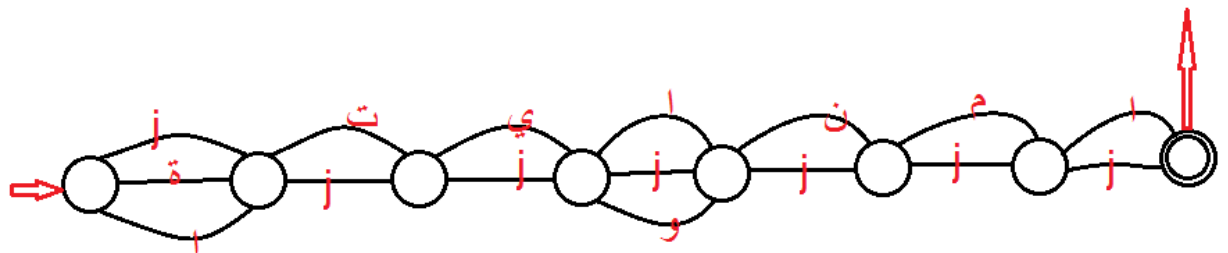
- Prefix (pref):



- Proclitics:



- Suffix:



Example:

?- parser('ذهب محمد الى المدرسة', P).

P = [[['المدرسة', [noun(مدرسة), [number, singular], [gender, female]], [nom(مدرسة), proc(ل), proc(إ), name], []], [['الى', [particule(الى)], [particule(p3(ي)), particule(p2(ل)), particule(p1(إ)), part], []], [['محمد', [noun(محمد), [number, singular], [gender, male]], [nom(محمد), name], []], [['ذهب', [verb(ذهب), [number, singular], [gender, male], [aspect, past]], [asl(a3(ب)), asl(a2(و)), asl(a1(ذ)), verb], []]]]