Distinguishing Nominal and Verbal Arabic Sentences: A Machine Learning Approach

Duaa Abdelrazaq, Saleh Abu-Soud and Arafat Awajan
Princess Sumaya University for Technology
Jordan

Abstract

The complexity of Arabic language takes origin from the richness in morphology, differences and difficulties of its structures than other languages.Thus, it is important to learn about the specialty and the structure of this language to deal with its complexity. This paper presents a new inductive learning system that distinguishes the nominal and verbal sentences in Modern Standard Arabic (MSA).The use of inductive learning in association with natural language processing is a young and an interdisciplinary collaboration field, specifically in Arabic Language.The resultsobtained out of this research are ambitious and prove that implementing inductive learning in Arabic complex structure will gain a promisingcontribution in the field of Arabic natural language processing (ANLP).

Keywords: Arabic LanguageProcessing, Natural Language Processing, Inductive Learning, ILA

1. Introduction

Machine learning (ML) with respect to Natural Language Processing (NLP)has gained more importance since the availability of increased number of electronic documents in the form of unstructured and semis structured by automatically classify and discover patterns from electronic documents. The increasing association between ML and NLP has been proved that most NLP problems can be viewed as classification problems (Magerman, 1995) and(Daelemans et al. 1997).

Inductive Learning determines a general rule from a set of empirical instances,this process specify *learning by example* form.Inductive learning algorithms (Quinlan, 1988), (Forsyth, 1989), (Hancox *et al.*, 1990) and(Michalski, 1983) are algorithms that generate specific data into general rules. These works prove that induction of rules from observed data is a useful technique for automatic knowledge acquisition.

The term "إعراب",<ErAb, (<ErAb in Buckwalter transliteration) in Arabic designates the primary constructing tool that is used to analyze each word in a sentence and determines if it is grammatically correct. The first step on the <ErAb is to distinguish the nominal and verbal sentences. The objective of our approach is to design a comprehensive system able todistinguish most of nominal and verbal sentence cases.However, we are not aware of any other work that chooses to model the learning knowledge for distinguishing nominal and verbal Arabic sentences. The majority researchers' investigations were revolving in particular cases for nominal or verbal sentences, and in a partial way using rule models as a sub-task in a complete process of linguistic analysis as (Ditters, 2000), (Othman et al., 2004) and (Hammadi, and Aziz, 2012).

2. ILA: The Inductive Learning Algorithm

In this paper, we apply an inductive learning algorithm called ILA, to distinguish between nominal and verbal sentences of Arabic language.

ILA, by (Tolun and Abu-Soud, 1998),createsa set of classification rules from training examples that proposed as a supervised learning approach. This inductive algorithm produces IF-THEN rules in an iterative modefrom a set of examples that has a single class. The algorithm depends on the generality of the database patterns and eliminates the unnecessary conditions. When a rule is being found, ILA marks the examples that covered the detecting rule and removes those examples from the training set.

ILA rules are more simple and general than those obtained from other known algorithms, as it has been proven in their work, in which the classification capability of ILA increases by the generality of the rules.

3. System Framework

Toapply machine learning on Arabic Text we have to transfer the unstructured text form to Tags structure database form.Toacquire this, a pre-processing phase must be applied, which appear on afterward as a complex task relative to the Arabic text structure.

The System framework passes through three main phases:Arabic sentences collection, preprocessing and Classification. This framework is shown in Figure 1.

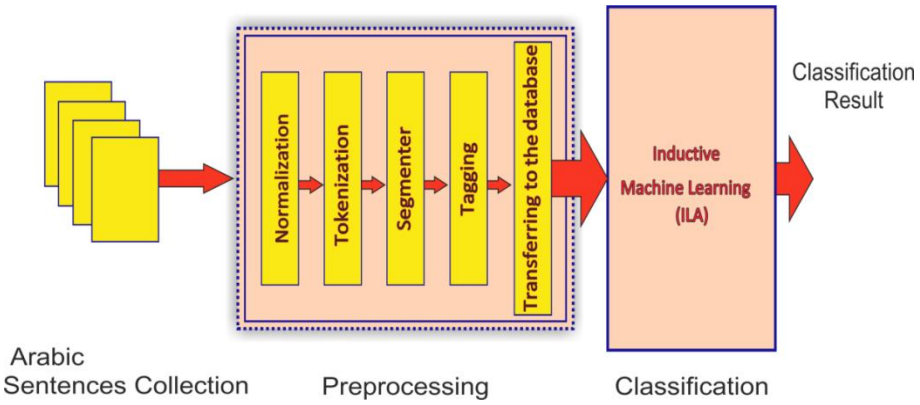

Figure 1. System Framework

These phases are described in the following points:

a) Arabic sentences collection phase

We collect the data set that will be used for building and testing the ILA module. We use 376 sentences, with unique structures, well formed and linguistically reviewed by Arabic linguistics specialist in Arabic Grammar.

The grammar of Arabic, contains the grammar knowledge that's required to analyze a sentence. Ideally, the grammar was being developed especially for the purposes of understanding the Arabic text.

According to *Albasria School* (المدرسةالبصرية), which provided a definition of two sentence types in the traditional Arabic grammar (إعراب) is that nominal sentences (الجملةالاسمية) begin with nominals (nouns, pronouns, etc.) and verbal sentences (الجملةالفعلية)

begin with verbs (Habash et al., 2007). This definition follows in addition the sentence headed by (انوأخواتها) as it also has been considering as a nominal sentence, and the sentence headed by (كانوأخواتها) considered as a verbal sentence, since this verb is considered to be incomplete (ناقص). For this, we will adopt and follow this common agreement definition for MSA sentences. Furthermore, sentencescan be subdivided into simple sentence, which is not connected by any means with another sentence, and a compound sentence, which is composedof simple sentences connected with a conjunction article (اداة عطف).

b) Data preprocessing phase

In this phase, sentences are processed and prepared to be used by the classification phase. This phase has five main sub-phases; Normalization, Tokenization, Segmentation, Tagging and Transferring to a database as indicated in Figure 1.

i. Normalizationof the collected sentences.

The diacritics in documents are very difficult;they increase the characters from 28 letters to more than 300 characters which are subjected to many complex Arabic grammars (Elbery, 2012). Thus, in this paper, we used un-diacritic normalized sentences dataset.

Normalization of words is done by the following choices:

1. Remove diacritics
2. Replace آأا with ا
3. Replace يwith ى
4. Replace ة with ه

ii. Tokenization

In tokenization, the white spaces as well as the punctuation marks areconsidered as the mainmarkers that separate words in Arabic language. Typically, this process removes the non-Arabic letters, numbers and punctuations.According to our domain which recognizes and handles the sentence as it appears in the text, and since the Penn Tagging is defining all types of words, even the punctuations, we define a sentence as a stream of words that followed by a full stop, question mark, exclamation mark or a comma. Thus, recognizing a sentence in a stream of words in the document,the aforementioned punctuations will be removed as they appear at the end of each sentence and as identifiers for sentence boundary.

Figure 2 presents an example of the tokenization phase by splitting each word and removing the full stop from the end of each sentence.

iii. Stemming -Light- Segmenter

In addition to affixes, Arabic texts are full of agglutinationwhich contains proclitic (e.g. Articles, prepositions, conjunctions) and enclitics (e.g. Linked pronouns; الضمائر المتصلة) with stems (called lexical forms) (Awajan 2015). Thus, a numerous ambiguities of decomposing textual forms conductto a significant ambiguity inthe part of speech tags of Arabic words.

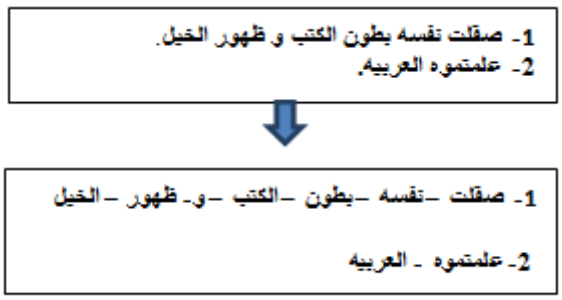

Figure2. Tokenization example

As it is mentioned earlier, since our approach appeals the structure of sentences that dedicates to our goal to distinguish the nominal and verbal Arabic sentences, a proper light-segmentation to each word is what matter, to enhance the output accuracy of the Stanford tagging, which uses the Penn Tags (Maamouri, 2004).In this paper, we will produce a light-stemming tool for recognizing suffixes and prefixes that indicate to all conjunctions, prepositions and pronouns (i.e. Thatrefers to some subject or object),which will be in-attached to the words but without any changing in the words form.

Substantially, each word will be segmented or in-attach any Suffix personal pronouns and Prefix preposition in order to improve the output of Stanford parser tagging set.Thus, the Sentence in **Arabic:** " صقلت نفسه بطون الكتب وظهور الخيل ",which is ("***Sqltnfsho bTwn Alktb wzhwr Alzyl*** " in Buckwalter transliteration), that corresponding in**Eng.: (Refined his soul the books and horseback).**

This sentence will be applied in the preprocessing phase and will be segmented to indicate to any Suffix or Prefix, that have not been recognized in the Stanford Parser Tags set, as it illustrated in Figure 3. Hence, the Determiner "ال" is already recognized by the Stanford Parser and it will not be separated.

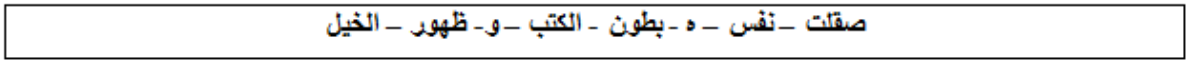

Figure 3. Light- segmenter

Tables 1, 2 and 3 present linked pronouns, prepositions and the prefix coordinating conjunction in Arabic respectively.

Table 1. The Suffix- linked pronouns- in Arabic

Number	Suffix	Number	Suffix
1	ك	11	ـا
2	ـه	12	ـوا
3	ـها	13	ـن
4	ـكم	14	ـي
5	ـكما	15	ـهما
6	ـنا	16	ـتا
7	ـكن	17	ـتن
8	ـهم	18	ـني
9	ـهن	19	ـنا
10	ـت		

Table 2. The Prefix- Preposition- in Arabic

Number	Prefix
1	كـ
2	لـ
3	بـ

Table 3. The Prefix- coordinating conjunction in Arabic

Number	Prefix
1	و
2	ف

iv. Part of speech – Tagging

The complexity that appears in the Arabic POS tagging is due mainly to different possible decomposition of a word and sentence. Since, a word in Arabic carried an inflection and clitics (e.g.Pronouns, conjunctions, and prepositions) (Mohamed and Kübler, 2010). Thus to solve the problem that will appear in the accuracy of the Stanford tagging, a segmentation-based approach has been previously adapted in the stemming phase.In this process, after transferring each sentence and their tokens -based segmenting-, we will use Stanford tagging that embedded on the Stanford parser to acquire proper accurate tags.

Thus, the Sentence ("***Sqltnfsho bTwn Alktb wzhwr Alzyl*** " in Buckwalter transliteration),English: (Polished his soul the books and horseback), **" صقلت نفسه بطون الكتب وظهور الخيل "** will be assigned as the following POS –tags:

Figure 4. The Stanford Tags

Figure 4 presents the Tagged sentence as produced by the online Stanford Parser. We should mention that we use the parser just to get the Penn Tags that have been used and embedded in the parser, and without using the parse tree (the grammatical analysis) of the given sentences.

v. Transferring to a data base

The representation of text will be modified in order to obtain words-POS-database. Thus, all Tags that correspond to each sentence will be transferred to a database.Furthermore, each sentence with its equivalent tags and the decision type of the sentence will be reviewed by Arabic linguistics in order to make sure that the structured sentences will be well suited to the learning process, that affect the entire learning system. Table 4 presents the previous sentence in the database with its equivalent Tags and the decision.

Table 4. The Equivalent Tags

	Tag1	Tag2	Tag3	Tag4	Tag5	Tag6	Tag7	Tag8	Tag9	Tag10	Tag11	Decision
1	VBD	NN	PRP$	NN	DTNN	CC	NN	DTNN	NA	NA	NA	V

c. Classification phase

In this phase, in order to classify the MSA sentences to nominal and verbal sentences, we adopt an inductive learning approach by choosing an Inductive machine Learning (ILA) algorithm (Tolun and Abu-Soud, 1998). The algorithm, as discussed earlier, produce an induction rules from a set of examples by generating IF-THEN rules. Substantially, the main advantage of the algorithm, that the rules are stated in an appropriate form for data exploration as well as the class description.

5. Implementation of ILA

We will explore ILA in ANLP field for distinguishing nominal and verbal MSA sentences. Typically, the presented supervised learning approach learns automatically from previously annotated sentences using training corpus –376 sentences as a dataset.

5.1 The examples

In our examples, the dynamic set of attributes is described in Tags with its own set of possible values, which have been obtained from the original corpus of Arabic POS tags with the parsed Arabic Treebank; these tags have been presented in Table 5.

Table 5. List of Penn POS tags used

NO	Tag	Discription
1	JJ	adjective
2	RB	adverb
3	CC	coordinating conjunction
4	DT	determiner/demonstrative pronoun
5	FW	foreign word
6	NN	common noun, singular
7	NNS	common noun, plural or dual
8	NNP	proper noun, singular
9	NNPS	proper noun, plural or dual
10	RP	particle
11	VBP	imperfect verb (***nb: imperfect rather than present tense)
12	VBN	passive verb (***nb: passive rather than past participle)
13	VBD	perfect verb (***nb: perfect rather than past tense)
14	UH	interjection
15	PRP	personal pronoun
16	PRP$	possessive personal pronoun
17	CD	cardinal number
18	IN	subordinating conjunction (FUNC_WORD) or preposition (PREP)
19	WP	relative pronoun
20	WRB	wh-adverb
21	,	punctuation, token is , (PUNC)
22	.	punctuation, token is . (PUNC)
23	:	punctuation, token is : or other (PUNC)
24	NOUN_QUANT	quantifier
25	ADJ_NUM	ordinal number

The examples are constructedfrom a sequence of -Penn POS- tagsthat represent the sequence of each annotated sentence which will be listed in a table.The rows correspond to thesentences and the columns contain attribute values of the Penn tagscorrespond to each word in the sentence. Finally, since the sentences length range are not the same we will indicate to each empty Tags that indicate the end of the sentence with NA attributes, dueto the space here reflects a valuable information (i.e. short, long, simple and compound sentences).

5.2 ILA algorithm

The process of applying a supervised Inductive Machine Learning to a document of Arabic sentences toward distinguishing the nominal and verbal sentences is illustrated in Figure 5.

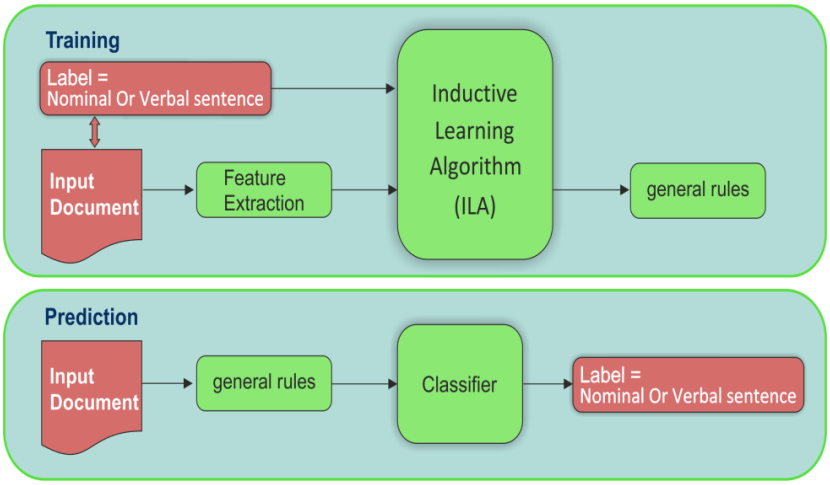

Figure 5.The process of supervised Inductive Learning

Once the classified document is labeled by verbal and nominal decisions and entered in the training dataset, we will extract features through applying ILA algorithm in order to extract general rules. Later, the induced rules will be applied on the test set in order to predict the nominal and verbal label decision for the unseen examples.The dataset contains of training examples, considering ***E***, each example, composed of ***A*** attributes and a class attribute**C** with two possible decisions (i.e. nominal and verbal) in our case.

5.3 Illustrative Example

As an illustrative example by implementing ILAoperations in our approach, we consider a training set, consisting of Fourteen examples (i.e. ***E***=14) with eleven attributes (***A***=11) and one decision (class) attribute with two possible values, {n, v}, (**C**=2). In this example, Tag i, i ϵ{1,2,3,4.5,6,7,8,9,10,11} are attributes of possible values of Penn Tags sets {e.g. VBD, VBP, DTNN, NN, etc.}, corresponding to their sequence located in the sentences.Table 6 presents the authentic sentences before the pre-processing phase, Table 7 represents the sequence of tags corresponding to the sentences given in Table 6.

Table 6.Sentences sample example data set

1	وهذه الاسباب لم تهيّئه للتعلّم.
2	إنّ قضيّة تعليم النّحو العربي للأجانب يشغل عقول المدرّسين.
3	درهم وقاية خير من قنطار علاج.
4	خير لك أن تتألّم لأجل الصدق.
5	تتضمن عملية التخطيط صياغة الاهداف التدريسية في صورة قابلة للتقويم.
6	ينام عميقاً من لا يملك ما يخاف من فقدانه.
7	أبي سافر إلى عمّان.
8	قوة السلسلة تقاس بقوة اضعف حلقاتها.
9	دقيقة الألم ساعة و ساعة اللذة دقيقة.
10	البستان الجميل لا يخلو من الأفاعي.
11	إنّه يعيش في بيئة غير عربيّة وفي مجتمع أجنبي.
12	فليس هناك وجه للمقارنة بين صنفين من الطلبة،
13	يحصد الفلاح القمح.
14	ركبت بالأمس زورقا مع أبي.

Table 7. Object Classification Training Set

NO	Tag1	Tag2	Tag3	Tag4	Tag5	Tag6	Tag7	Tag8	Tag9	Tag10	Tag11	Decision
1	CC	DT	DTNN	RP	VBD	PRP	IN	DTNN	NA	NA	NA	n
2	RP	NN	NN	DTNN	DTJJ	IN	NN	VBP	NN	DTNNS	NA	n
3	NN	JJ	NN	IN	NN	NN	NA	NA	NA	NA	NA	n
4	NN	IN	PRP	RP	VBP	IN	NN	DTNN	NA	NA	NA	n
5	VBP	NN	DTNN	NN	DTNN	DTJJ	IN	NN	NN	IN	DTNN	v
6	VBP	NN	WP	RP	VBP	WP	VBP	IN	NN	PRP	NA	v
7	NN	VBD	IN	NNP	NA	NA	NA	NA	NA	NA	NA	n
8	NN	DTNN	NN	IN	NN	JJR	NNS	PRP	NA	NA	NA	n
9	NN	DTNN	NN	CC	NN	DTNN	NN	NA	NA	NA	NA	n
10	DTNN	DTJJ	RP	VBP	IN	DTNN	NA	NA	NA	NA	NA	n
11	RP	PRP	VBP	IN	NN	NN	JJ	CC	IN	NN	JJ	v
12	CC	VBD	DT	NN	IN	DTNN	RB	NNS	IN	DTNN	NA	v
13	VBP	DTNN	DTNN	NA	NA	NA	NA	NA	NA	NA	NA	v
14	VBD	PRP	IN	DTNN	NN	IN	NN	PRP	NA	NA	NA	v

Since ***C***is two, **{n, v},** (***C***=2), the first step of the algorithm divided the examples in the two sub-tables which are shown in Table 8,*One table for* each possible value of the class attribute- **nominal and verbal- {n, v}.**

Table 8. Sub-Tables - According to Decision Classes Partitioned.

Sub-Table A

Example no. old	new	Tag1	Tag2	Tag3	Tag4	Tag5	Tag6	Tag7	Tag8	Tag9	Tag10	Tag11	Decision
1	1	CC	DT	DTNN	RP	VBD	PRP	IN	DTNN	NA	NA	NA	N
2	2	RP	NN	NN	DTNN	DTJJ	IN	NN	VBP	NN	DTNNS	NA	N
3	3	NN	JJ	NN	IN	NN	NN	NA	NA	NA	NA	NA	N
4	4	NN	IN	PRP	RP	VBP	IN	NN	DTNN	NA	NA	NA	N
7	5	NN	VBD	IN	NNP	NA	NA	NA	NA	NA	NA	NA	N
8	6	NN	DTNN	NN	IN	NN	JJR	NNS	PRP	NA	NA	NA	N
9	7	NN	DTNN	NN	CC	NN	DTNN	NN	NA	NA	NA	NA	N
10	8	DTNN	DTJJ	RP	VBP	IN	DTNN	NA	NA	NA	NA	NA	N

Sub-Table B

Example no. old	new	Tag1	Tag2	Tag3	Tag4	Tag5	Tag6	Tag7	Tag8	Tag9	Tag10	Tag11	Decision
5	1	VBP	NN	DTNN	NN	DTNN	DTJJ	IN	NN	NN	IN	DTNN	V
6	2	VBP	NN	WP	RP	VBP	WP	VBP	IN	NN	PRP	NA	V
11	3	RP	PRP	VBP	IN	NN	NN	JJ	CC	IN	NN	JJ	V
12	4	CC	VBD	DT	NN	IN	DTNN	RB	NNS	IN	DTNN	NA	V
13	5	VBP	DTNN	DTNN	NA	NA	NA	NA	NA	NA	NA	NA	V
14	6	VBD	PRP	IN	DTNN	NN	IN	NN	PRP	NA	NA	NA	V

We will initialize the attributes combination count; j=1 in the first sub-table. The list of attribute combinations comprises: Tag i, i ϵ{1,2,3,4.5,6,7,8,9,10,11}.
ILA divides the attributes into distinct combinations with j distinct attributes.For theexamining combination {Tag1} with its attribute value "NN", it appears in **sub-table A** but not in **sub-table B** with five times appearances.As a result, the maximum number of occurrences is "NN" -thefirst combination- will becalled **max-combination**. The attribute values "RP" and "CC" will not be considered in this step since they appear in both **sub-table A** and **sub-tableB**. For combination {Tag2} we have "DT, JJ, IN, DTJJ" with same occurrence of one time, for this case any value can consider as the maximum – combination for that attribute. For {Tag3}, we have "NN" with four occurrences and "PRP", "RP" with one occurrence. Consequently, "NN" value will be as the maximum- combination for {Tag3}. Continuing further with the combination for Tag i, i ϵ{1,2,3,4.5,6,7,8,9,10,11}. After examining all combinations, we found that {Tag1} with the attribute value of "NN" will be selected as the maximum of the maximum – combination (i.e. maximum number of occurrences) between all attributes because it has the maximum appearance for all Tags .

Thus, since the value of max-combination is recurrent in 3,4,5,6 and 7 rows, those rows will be marked as classified in sub-table A and the following rule (Rule 1) will be extracted:

- **RULE 1**: **If Tag1 = NN => N**

These steps will be applied over the remain unmarked examplesin sub-table A (i.e. rows 1, 2 and 8). Thus, "DTNN" is the only attribute value of {Tag1}- since we eliminate "RP" and "CC" for being appears in sub-table B. "DT" and "DTJJ" the attribute value of {Tag2}, andso on, continuing examining the attribute values fromTag3 toTag11. As it is shown in Table 8, the number of occurrences is the same for all remaining attributes (i.e. each occurring once). Thus,the first occurrence'sattribute by default will be selected (i.e. "DTNN" attribute value of {Tag1}). Rule 2 is appended to the rule set and the eighth row in sub-table Awill be marked as classified:

- **RULE 2: If Tag1 = DTNN => N**

For the remaining rows, first and the second unmarked remaining rows will be induced rules as follow:

- **RULE 3: If Tag2 = DT => N**

- **RULE 4: If Tag3 = NN => N**

By marking the first and the second remaining rows, the whole first sub table -which indicate the nominal sentences- is now classified, we will progressthe same steps on sub-table B.

In the second sub table, "VBP" is the attribute value of {Tag1} appears three times in the first, second and fifth rows in sub-table B. Thus, the three rows will be classified and Rule 5 will be added to the rule list.

- **RULE 5: If Tag1 = VBP => V**

Now, we have row 3,4 and 6, but regarding to the algorithm, we will exclude Tag1 for the row 3 and 4, since Tag1 in these rows was appeared in the first sub table. Thus, the only choice for {Tag1} is the attribute with the value of "VBD",which appears in the row 6, itwill be marked as classified and the sixth rule will be extracted:

- **RULE 6: If Tag1 = VBD => V**

For row 3, the number of occurrences is the same for all remaining attributes (i.e. each occurring once). The algorithm applies and selects the first one by default (i.e. "PRP" attribute value of {Tag2}). Rule 7 will also be extracted and inserted to the rule set:

- **RULE 7: If Tag1 = PRP => V**

For the last remain row (i.e. row 4), the occurrences are the same for all remaining attributes (i.e. each occurring once). The algorithm applies and selects the first one by default (i.e. "PR" attribute value of Tag3). Thus, the row will be marked as classified and Rule 8 (i.e. the last rule) will be added to the rules list:

- **RULE 8: If Tag3 = PR => V**

All of the rows of sub-table B are marked as classified, the algorithm terminates and exits with the set of rules extractedto this point.

Furthermore, as it has been noticed that the entire previous example does not fit, if the max-combination = ϕ. In that case, according to ILA j will be increased by 1. **Table9** presents the case of one remaining row (i.e. the first row in Sub-Table D), which each attribute values appear in both sub tables (i.e.max-combination = ϕ). The algorithm willincrease j by 1 and generate combinationsof 2 attribute, {Tag1 and Tag2}, {Tag1 and Tag 3}, {Tag 1 and Tag 4} and so on, examining all the possibility. Hence, the "CC and PRP" value of {Tag 2 and Tag 3} combination is disregarded because it appears in sub-table C, and this isapplied on combinations of the same conditions. The first and second combinations suit the conditions as they both appear in sub-table D but not in sub-table C for the same attributes. Thus,{Tag1 and Tag2} will be selected. Consequently, the following rule is obtainedand the row is marked as classified:

- **IF Tag1= CC and Tag2= RP => V**

Sub Table C

Tag1	Tag2	Tag3	Tag4	Tag5	Tag6	Tag7	Tag8	Tag9	Tag10	Tag11	Decision
CC	NN	PRP	RB	NOUN_QUANT	NN	NA	NA	NA	NA	NA	n
CC	NN	PRP	VBP	DTNN	IN	DTNN	IN	DTNN	NA	NA	n
CC	DTNN	DTNN	NN	IN	NNP	NA	NA	NA	NA	NA	n
CC	NN	DTNN	NN	PRP	JJ	NA	NA	NA	NA	NA	n
CC	DT	DTNN	VBP	RP	VBP	NN	NN	NN	DTNN	NA	n
CC	DT	IN	RP	PRP	VBP	NN	NN	DTNN	NN	NA	n
CC	IN	NN	DTNN	CC	RP	NN	DTNNS	IN	DTNN	NN	n
CC	NN	DTNN	IN	NN	DTNN	IN	NN	DTNN	NA	NA	n
CC	NN	PRP	NNS	VBP	IN	DTNN	NA	NA	NA	NA	n
CC	DT	DTNN	RP	VBD	PRP	IN	DTNN	NA	NA	NA	n
DTNN	RP	PRP	NN	NA	NA	NA	NA	NA	NA	NA	n

Sub Table D

CC	RP	PRP	RP	VBP	NN	DTNN	NA	NA	NA	NA	v
CC	VBP	NNP	IN	NN	VBP	NN	PRP	RB	NN	DTNN	v
CC	VBP	RB	DT	IN	NN	DTNN	IN	NN	PRP	DTJJ	v
CC	VBD	IN	NN	NN	DTNN	DTJJ	IN	NNS	NNS	PRP	v

Table9. The case of row's attributes that appear in both sub tables.

6. Experiments and Results

In this section, we will evaluate the proposed system to assess its accuracy, through a series of experiments on 376 well annotated (i.e. Gold Standards) Arabic sentences that range from 2 to 11 words, whichpresent simple to complex MSA sentences.

This induction system inserts a set of Arabic training sentences as a file of ordering attribute values set, labeled by a decision attribute for each example. The results are created as a set of individual rules for each of the classificationdecision attributes.

The evaluation of a learning system is being assessed by the accuracy of the classification rules on unseen examples.

The proposed system will load the database sentences in the memory in order to divide it into two sets, training sentences and test sentences. Thus, after the database has been loaded, it will randomly select the test sentences from the dataset as unseen examples. Consequently, the remaining of the dataset will establish the training set in which the algorithm will run and generate the inductive rules. Subsequently,the unseen examples will obtain and predict the decisions class label. For enhancing the generality of the evaluation results, the test had been randomly conducted on the exceeding cases four times, containing the training sentences and the unseen sentences as well.

To evaluate the proposed system, four experiments have been conducted; each is repeated four times with different portions and different number of sentences as follows: 125, 188, 250, and 376 sentences respectively. Table 10 shows the obtained results by applying the system on different number of sentences, with different percentages of unseen examples. As shown on the table, we got around 89.47% accurate for 15% unseen examples on 125 sentences, while for 75% unseen examples we got around 74.31% accuracy which is not bad for a small number of sentences. However, it is noticed that the accuracy improves as number of sentences increases. Even that the results are domain-dependent and affected by the randomness of unseen sentences, the best results are almost obtained from 376 sentences. Figure 6 summarizes these results in more clearly.

Table10. Results obtained for different number of sentences

% of Unseen examples	Average Accuracy			
	125 sentences	188 sentences	250 sentences	376 sentences
15%	89.47	87.58	87.89	92.63
25%	85.62	85.95	90.47	91.48
50%	75.15	83.19	85.44	88.93
75%	74.31	78.58	78.93	78.72

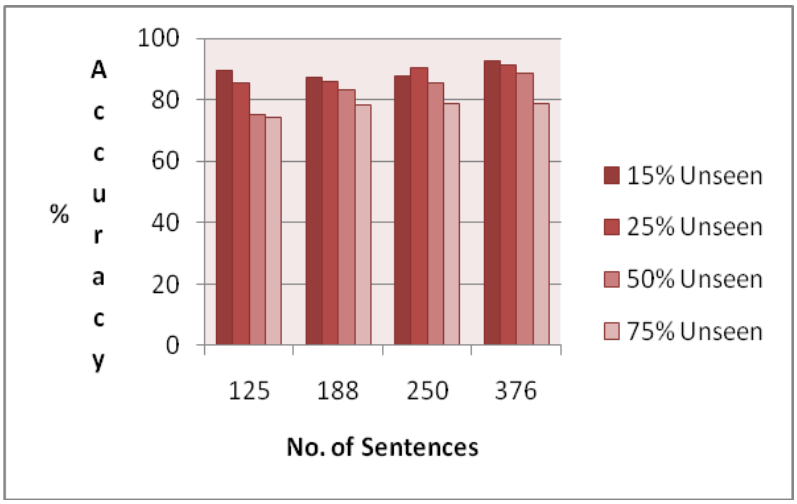

Figure 6. Average accuracy of the system for different number of sentences with different number of unseen examples

In another experiment, the precision of nominal and verbal sentences is calculated as it defined by the equation (1) below.

Precision = TP/(TP+FP) …… (1)

Where TP = True Positive
FP = False Positive

Tables 11 and 12 show the average results of 5 experiments, each is repeated 5 times, for10% of unseen sentences (i.e. 38 sentences) from the total of 376 sentences for Nominal and Verbal sentences respectively.

Table 11. Precision for Classifying Unseen Nominal Sentences.

Experiment	TP	FP	Precision
1	12	4	%75
2	15	4	%79
3	16	2	%89
4	9	9	%50
5	12	4	%75

Table 12.Precision for Classifying Unseen Verbal Sentences.

Experiment	TP	FP	Precision
1	21	1	%95
2	19	0	%100

3	19	1	%95
4	20	0	%100
5	22	0	%100

As it is shown, the system behaves extremely well in predicting the verbal decision than the nominal decision. This is because; from the point of view of Arabic linguists, the nominal sentences basically have a lot of structures than the verbal sentences. Consequently, the nominal decision is more difficult to predict due to their ambiguity structure characteristics.

9. Conclusions

This paper has been carried out with the aim of understanding Modern Arabic sentence structure. The main objective of this work was to implement an inductive machine learning approach to distinguish the nominal and verbal Arabic sentences for MSA. To achieve our goal, ILA as an inductive learning algorithm induced a set of rules based on the training set from a structured data text. Therefore, the sentences have been preprocessed, and structured, through tagging the words sequence in sentences using the Stanford tagger that impeded in their parser. The accuracy of Stanford tags in their online parser tool is not appropriate for the learning process nor have appropriate segmentation. For this reason, in our framework we produced a semi segmenter to separate any clitic that indicates to any subject or object. Furthermore, the sentences have been rechecked and corrected by Linguists in the stemming and in the tagging phase to obtain Gold standard sentences.

The inducted rules dealt with different sentence structures, word agreement and ordering problem. The results shown in Table 10 proved that ILA has provided a very good accuracy results with 92.63 in 15% unseen sentences. Furthermore, we have measured the precision of the nominal sentences decisions, as well for the verbal sentences in the unseen test set. The results, as shown in the Tables 11 and 12, the system is more accurate in predicting the verbal decision than the nominal decision. According to the point view of Arabic linguists for that case, the reason is that the nominal sentences basically have many structures than the verbal sentences.

Finally, the successful results of this research show that ILA is general and robust algorithm for applying it in the field of ANLP.

11. References

Awajan A. 2015. Keyword Extraction from Arabic Documents using Term Equivalence classes. ACM Transactions on Asian and Low-Resource Language Information Processing. Volume 14, Issue 2, Article 7.

Buckwalter, T. (2004), Buckwalter Arabic morphological analyzer version 2.0.

Daelemans, W., Van den Bosch, A. and Weijters, A. (1997), Empirical Learning of Natural Language Processing), Sandstorm to make internet clearer for Arabic users - IT Business . www.itp.net/486224-sandstorm-to-make-internet-cleare.

Ditters, E. (2000), A Formal Grammar for the Description of Sentence Structure in Modern Standard Arabic, TCNMO, Nijmegen University, The Netherlands.

Elbery, A. (2012), Classification of Arabic Document, CS 5604 : Information Storage and Retrieval, ProjArabic team.

Forsyth, R. (1989), Machine Learning principles and techniques, Ed: R. Forsyth, Chapman and Hall, London.

Habash, N., Gabbard, R., Rambow, O. and Kulick, S. (2007), Determining Case in Arabic: Learning Complex Linguistic Behavior Requires Complex Linguistic Features, Center for Computational Learning Systems, Columbia University.

Hammadi, O.I. and Aziz, M.J.A. (2012), Grammatical Relation Extraction in Arabic Language, School of Computer Science,Faculty of Information Science and Technology,University Kebangsaan Malaysia, 43600, Selangor, Malaysia.

Hancox P. J., Mills W. J, Reid B. J.,(1990), "Artificial Intelligence / Expert System", Kansas.

Maamouri, M. and Bies, A. (2004), Developing an Arabic Treebank: Methods, Guidelines, Procedures, and Tools. In Proceedings of COLING 2004. Geneva, Switzerland.

Magerman, D. (1995), Statistical decision tree models for parsing. Proceedings of the 3rd annual meeting on ACL, 1995. Bolt Beranek and Newman Inc. 70 Fawcett Street, Room 15/148 Cambridge, MA 02138, USA.

Michalski, RS.(1983), A theory and methodology of inductive learning, Artificial intelligence - Elsevier.

Mohamed, E. and Kübler, S . (2010), Arabic Part of Speech Tagging -,Indiana University. , Department of Linguistics Memorial Hall 322 Bloomington, IN 47405 USA.

Othman, E., Shaalan, K. and Rafea, A. (2004), TOWARDS RESOLVING AMBIGUITY IN UNDERSTANDING ARABIC SENTENCE, International Conference on Arabic.

Quinlan, J.R. (1988), Induction, knowledge and expert systems, in Artificial Intelligence Developments and Applications, Eds J.S. Gero and R. Stanton, Amsterdam, North-Holland, pp.253-271.

Tolun, M.R. and Abu-Soud, S.M. (1998), ILA: an inductive learning algorithm for rule extraction, Expert Systems with Applications, April 1998, Pages 361–370.