Available at: www.sabauni.net/ojs

S.J.I.T.N Saba Journal of Information Technology and Networking

Article

N-Attributes Stochastic Classifier Combination for Arabic Morphological Disambiguation

Dr.Mohammad Albared ,Dr.Muneer Hazaa

Thamar University Faculty of Computer Science and Information System, Yemen

Article info

Article history:

Accepted Jan, 2015

Keywords:

Morphological disambiguation,

Classifiers combination,

N-attributes

Abstract

Morphological disambiguation is the ability to computationally determine which morphological tag of a word is activated by its use in a particular context. The main problem in statistical morphological disambiguation of rich morphological languages is data sparseness, where the level of ambiguity is high and the potential tagset size is very large. This paper investigates several fully supervised stochastic morphological disambiguation approaches for morphologically rich languages, with a specific application to Arabic. First, this paper evaluates the direct statistical disambiguation method in which only one tagging model is used. In this approach, each word is assigned a complex morphological tag. In addition, this paper introduces the single-attribute classifiers combination method in which the problem is decomposed into several single-attribute disambiguation sub-problems. Then, a classifier combination method, which consists of several trigrams HMM tagging models and a module which combines them, is used. Results show that the first method suffers from data sparseness and has large tagging time and the second one has low tagging accuracy. Finally, the paper present a novel approach based on the combination of several N-attributes morpheme-based probabilistic classifiers. First, the morphological disambiguation problem is decoupled into several N-attributes tagging sub-problems. Then, several classifiers are used to solve each sub-problem. Finally, the outcomes of all N-attributes classifiers are combined. Several problem decomposition methods and classifiers combination algorithms are investigated. The triple-attributes (N=3) stochastic classifier combination model provides an overall tagging accuracy of 91.5%, reduce the data sparseness problem and saves run time over the direct approaches.

Corresponding author:Dr. Mohammad Albared ,
E-mail address: alialbared@gmail.com

1. Introduction

Morphological Disambiguation, also known word-class syntactic tagging or fine-grained part of speech (POS) tagging [1-3], is an intermediate layer between morphological and syntactic analysis in which each word appearing in a text is assigned an unambiguous morphological tag [4]. It can also be defined as the ability to computationally determine which morphological tag of a word is activated by its use in a particular context. In morphological disambiguation, the POS tag set is more fine-grained and defined in terms of morphological and grammatical attributes (features) characterizing word structure. Tagging for POS alone would not solve the morphological disambiguation problem[5]. With the term morphological tag, we mean the morphological attributes of a word such as POS, person, number, gender and tense. Tagging with fine-grained tag set is an important step for many NLP tasks such as syntax parsing, word sense disambiguation, semantic parsing analysis and language modeling for speech recognition.

Arabic, like other Semitic languages, has rich inflectional, derivational and templatic morphology [6-8]. The main important qualitative distinction between the typical POS tagging in simple languages and the morphological disambiguation is the large number of possible tags that can be assigned to a word. Unfortunately, the high number of possible tags poses data sparseness challenge for the typical statistical models[1]. A large and fine-grained tag set may enrich the supplied information but the performance of the tagging model may decrease. Morphological ambiguity is very difficult problem in Arabic and some other languages (Levinger, et al., 1995). Thus, finding methods to reduce the morphological ambiguity in the language is a great challenge for NLP researchers . In fact, a much richer model is required to be designed to capture the encoded information when using a fine grained tag set and hence, it is more difficult to learn [9]. In addition, the situation becomes more serious in a low-resource setting.

In general, most previous studies have been treated the morphological disambiguation problem either as a single classification problem and tackled all morphological and grammatical attributes together in one step or as a combination of multiple classification sub-problems, each one is a single morphological attribute classification problem. In the first approach, each word or sub-word (morpheme) in the training data is labeled with a composite tag which can be mapped into a vector of values of grammatical categories"<----------->"[10]. Each morphological and grammatical attribute (morphological properties) has slot "-"of this vector (positional notation). The first slot is usually filled by one of the POS attribute values. Each one of the remaining slot can only contain one of its morphological attribute values or null"-" if its morphological attribute is irrelevant for the POS value in the first slot. The tag set consists of all possible combination of the values of these morphological and grammatical

attributes. The potential tag set size is very large especially with morphology rich languages. However, this approach suffers from the data sparseness. In fact, to ensure statistical significance, a very large training data are required. Even if the training data are very large, it is impossible to avoid the problems of data sparseness and out of word-tag pairs i.e. not all words in the training data have all their possible tags in the training data. Moreover, word-tag or morpheme-tag pairs that appeared with a statistically insignificant frequency will be assigned a poor probability estimate.

In the second approach, the morphological disambiguation problem is decomposed into m disambiguation sub-problems. This is done by dividing the training data into m training data. Although each of these data contains the same words, every one of them is tagged using a different morphological attribute. Then, a classifier combination method, which consists of several simple classifiers and a module which combines them, is used. In fact, this method does not suffer from data sparseness. However, it is less accurate than direct method and it requires a sophisticated combination algorithm to re-impose the linguistic dependencies that is lost during the problem decomposition[11, 12].

In this paper, we evaluate, compare and contrast these two approaches in morphologically rich languages, with a specific application to Arabic. Moreover, this paper proposes a new approach to the morphological disambiguation problem. This method is also based on dividing the morphological disambiguation problem into multiple classification problems. Unlike previous work, the morphological disambiguation problem is divided into several N-attributes classification problems to balance between the sparseness problem and the loss of the linguistic dependencies. Then, each classification problem is solved independently using a trigram HMM classifier. After that, a combination algorithm is used to combine the intermediate results of these classifiers to generate the final result.

The paper is organized as follows: In Section 2, we will review the previous approaches to the morphological disambiguation problem. In Section 3, we present relevant linguistic properties and the morphological ambiguity in Arabic. In Section 4, we introduce the data used and the morphological tag set. In Section 5, we formalize the morphological disambiguation problem in statistical context. In Section 6, we introduce our n-attributes classifier combination approaches and give our experimental results. Finally, we conclude in Section 7.

1.1. Related Works

Several works have been developed for morphological tagging of agglutinative or highly inflectional languages such as Turkish, Hebrew and German. In the case of language where the morphology is simple, morphological disambiguation is generally covered under the task of simple POS tagging. The main morphological attributes are embedded in the tag

name (for example, Ns and Np for noun singular or plural). In this section, we review related work on morphological tagging for morphologically rich languages. Turkish words have been actively studied since the seminal work from Oflazer and Kuruoz [13] that used constraint-based approach with hand crafted rules for Turkish morphological disambiguation. They select the right morphological tag based on local neighborhood constraints, heuristics and limited amount of statistics.

Arabic morphological disambiguation has not been studied extensively, especially using statistical approaches. In fact, the main reason is the lack of free publicity morphologically annotated corpora. Recently, a morphologically annotated version of the Quranic Arabic Corpus has become freely available through the Quranic Arabic Corpus project[14]. This facilitates, at least to some extent, the application of advanced machine learning techniques to the problem of Arabic morphological disambiguation. However, Habash and Rambow [15] and Smith et al.[16] worked to address the problem of Arabic morphological disambiguation. These two works utilize Buckwalter Arabic Morphological Analyzer (BAMA). Their approaches are based on disambiguation the output of BAMA i.e. they choose among a limited number of possible tags given by BAMA instead of considering all possible tags that a word may take. However, only Arabic words which can be analyzed by BAMA can be disambiguated. If the correct output is not enumerated by the morphological analyzer, it cannot be predicted.

In this paper, several statistical morphological disambiguation approaches are investigated for morphologically rich languages, with a specific application to Arabic. All these approaches are fully supervised, stochastic and dictionary-free. The proposed methods only uses morphologically-tagged corpus as an information source, and can automatically acquire a knowledge base from this corpus. It can be applied not only to Arabic language but also to other morphologically rich languages. This study first compare and contrast direct classification method and single attribute classifiers combination method. Furthermore, this work also designs a better method to handle the morphological disambiguation which balances between these two approaches. In this method the morphological disambiguation problem is decomposed into several N-attribute classification problems. The main idea of this method is to avoid the data sparseness problem and to retain the linguistic dependencies between the morphological attributes.

1.2. Data and the morphological tag set

In this work, the data used is the morphologically annotated version of the Quranic Arabic Corpus[14, 17]. In this version, each word is annotated with its full morphological information including the POS tags of its morphemes. The Quranic Arabic Corpus is an annotated linguistic resource which shows the Arabic grammar, syntax and morphology for each word in the Holy

Quran, the religious book of Islam which is written in classical Quranic Arabic (c. 600 CE). The Quranic Arabic Corpus consists of 77,430 words of the Quranic Arabic. In fact, we have used a morpheme-based version which consists of about 128219 tokens. The Qur'an is always published in fully vocalized versions. For the purpose of this work, we use fully unvocalized versions. All vowels are removed. Nowadays, Arabic is written mostly in unvocalized script. Vowels (diacritics) are no longer used in printed or electronic Arabic text. Unvocalized Arabic texts are more ambiguous than vocalized or partially-vocalized Arabic text. We have chosen the Quranic Arabic Corpus for training and testing our models, since it is the only free available morphologically annotated Arabic corpus. However, our proposed models are language, data and tag set independent i.e. they can also be applied for any language, data and any tag set.

The tag set (morphological tag set), used in this work, consists of multiple dimensions tags. In addition to POS tags, multiple morphological features are assigned to each morphological segment.

In the original version, morphological tags are represented in concatenative forms. In this concatenative representation, only the relevant morphological features are shown. Moreover, the default value of some morphological features is not shown such as the definiteness marker "Def". However, as in [10], we represent morphological tags in slots form (-.-.-.-.-.-.-.); a tag consists of 7 slots separated by dots; each position (slot) represents a feature and the tag value (letter, number or three letters) at that location represents a value or attribute of the morphological feature. The dash "-" represents a feature not relevant to a given word. The first slot shows the main POS. The slots 2, 3, 4, 5, 6 and 7 are used to represent other morphological features: person, gender, definiteness, number, tense and voice, respectively.

1.3. Problem Definition: The Statistical Morphological Tagging

In this paper, the Arabic Trigram HMM tagger [18, 19] is used as the basic classifier in our environment. In this context, the morphological disambiguation problem is formally defined as follows: given a sequence of words, $w_1^s = w_1, ..., w_s$, find the best sequence of morphological tags $mt_1^s = mt_1, ..., mt_s$ of the words. This can be formulated as follows:

$$mt_1^s = \arg\max_{t_1^s} \prod_{i=1}^{s} p(mt_i | mt_{i-1}, mt_{i-2}) . p(w_i | mt_i)$$

MT is the morphological tag set

F_1 is the set of all POS tags

$F_2 = \{1, 2, 3, -\}$

$F_3 = \{m, f, -\}$

$F_4 = \{def, undef, -\}$

$F_5 = \{s, d, p, -\}$

$F_6 = \{perf, impf, impv, -\}$

$F_7 = \{PASS, ACT, -\}$

The simplest way to compute the parameters for the HMM is to use relative frequency estimation, which is to count the frequencies of word/tag and tag/tag. This way is called maximum likelihood estimation. However, data tend to be sparse especially with the large size of the morphological tag set and the small size of the training data. Due to this, the transition probabilities are smoothed using the linear interpolation of unigram, bigram and trigram maximum likelihood estimates in order to estimate the trigram transition probability.

In any tagging system, we frequently encounter words that do not appear in training data especially when only a limited amount of training material is available. The existence of such words is one of the main problems in any tagging system, since the statistical information of these words is unavailable. Unknown words are usually handled by an exceptional processing. Accuracy of unknown words is usually much lower than that for known words.

For handling unknown words, the Arabic HMM tagger uses a lexical model based on the linear interpolation of both word suffix probability and word prefix probability. Briefly, the model estimates the lexical probabilities of unknown words as follows: Given an unknown word w, the suffix probabilities $P(suffix(w)|mt)$ are estimated using the suffix guessing algorithm as described in (Brant,2000). Then, the lexical probabilities $P(prefix(w)|mt)$ are also estimated using the same way, but, the letters in the words are reversed before adding them to the new word tree in order to find the prefix probability. Finally, we use the linear interpolation of both the lexical probabilities obtained from both the word suffix and prefix to calculate the lexical probability of the word w as in the following Equation:

$$P(w|mt) = \lambda P(suffix(w)|mt) + (1-\lambda)P(prefix(w)|mt)$$

where λ is an interpolation factor.

1.4. N-Attribute Classifiers Combination Algorithm

Our methods for morphological disambiguation problem are based on the decomposition of the M N-attributes disambiguation sub-problems by grouping the morphological attributes into M groups. Each group contains N attributes. These methods have to avoid the data sparseness problem and to retain the linguistic dependencies between the morphological attributes.

However, the problem decomposition and m-attribute classifiers combination algorithm is as follow (see Figure 1):

1. First, group the morphological attributes $\{F_1, F_2, F_3, F_4, F_5, F_6, F_7\}$ into M groups $\{G_1, G_2, ..., G_M\}$, where $1 \le M \le 7$ and $G_i \neq G_j$ if i $\neq$ j. Each group contains N morphological attributes.
2. Second, the problem is decomposed into M tagging sub-problems. To do so, M $\{D_1, D_2, ..., D_M\}$ different training data are generated from the training corpus. All training data contains the same words. Each

training data D_i is tagged with the possible combination of the morphological attributes of group G_i.

3. Third, M classifiers $\{C_1, C_2, ..., C_M\}$ are used, each classifier is a trigram HMM tagger, and each classifier C_i is trained using D_i.
4. Thirdly, each classifier is tested using a test set that is tagged using the same tag set as in the data used to train it. Each classifier predicts only small set, N, of the seven morphological attributes. Every classifier produces its own intermediate result which is annotated using the same annotation scheme of the training data used to train this classifier.
5. Finally, the intermediate results are combined using a combination algorithm to produce the final result.

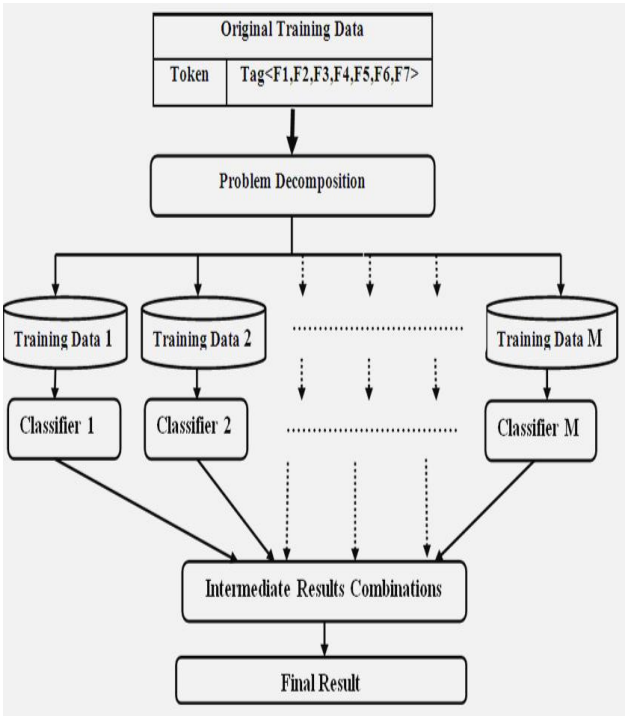

Fig. 1- Problem decomposition and classifiers combination algorithm

2. Evaluation

In the following subsections, we will first evaluate typical HMM POS tagging models for Arabic fine-grained POS tagging (direct classification method) for morphological disambiguation problem. Then, we will introduce several methods of morphological problem decomposition and classifiers combination and evaluate them. In all the experiments described in this paper, the term classifier refers to the Arabic trigram HMM tagging model. In addition, the data sets are split into two sets; training set 89.1% and test set 10.9 %.

2.1. Direct Disambiguation Methods

According to Equation (1), standard POS tagging techniques can be used directly to handle the morphological disambiguation problem. However in this experiment, we directly train and test both

Arabic HMM tagger with the linear interpolation guessing model and the TnT tagger using the morphologically annotated Arabic Quranic Corpus (M=1). Each word is tagged with a composite tag (N=7). The results of both tagging models are shown in Table 1. The results show that Arabic Trigram HMM outperforms the TnT Tagger. However, the testing time (tagging time) of both taggers is so large due to the tag set size. Essentially the time complexity of trigram HMM-based tagger is $O(n \times T^3)$ [20], where n is the number of words in the target sentence and T is the size of tag set. In this case, the direct method for morphological disambiguation problem T is very large, $T = |MT| \subset \{F_1 \times F_2 \times F_3 \times F_4 \times F_5 \times F_6 \times F_7\}$. The major factor that affects the performance of HMM-based tagger becomes the size of the tag transition probability set. The larger the size of training probability set or tag set, the more time it will take. Therefore it is noticed that when V is the size of vocabulary, and $T = |MT|$ is the number of tags, the space complexity for trigram HMM-based tagger is $O(V \times T^3)$.

Table 1- Results of direct classification methods

Model	Unknown %	Overall %	Training Time	Testing Time
TnT Tagger	33.3	90.33	0.30	23.25
Arabic Trigram HMM with the Linear Interpolation	48.7	91.03	0.30	23.37

2.2. Single Morphological Attribute Classifiers Combination

In this experiment, seven (M=7) new training data {D_1, D_2,…, D_7} are created from the original training data. Each one is tagged with the possible values of only single attribute (N=1). The first one is tagged with all POS tags, the second one is tagged with all Person tags {1, 2, 3,-} and so one. Each training data has its own tag set. Then, we use seven trigram HMM classifiers {C_1, C_2,….., C_7}. Each classifier C_i is trained with training data D_i. After that, each classifier is tested using the same test set. Each classifier predicts a small set of possible values of one morphological attribute. The first one is used to predict POS tags; the second one is used to predict person tags and so on. Using the experiment setup in the previous section, the intermediate results of all the simple classifiers are shown in Table 2. As shown in the table, the tagging performance of the trigram HMM classifier (all the classifiers are trigram HMM classifiers.) varies among the morphological attributes. The classifier achieves the highest overall tagging accuracy (98.13%) on voice attribute and the lowest overall tagging accuracy (94.27%) on POS attribute. It is also interesting to note that the unknown word handler algorithm (the linear interpolation guessing algorithm), which has achieved considerable success in guessing the POS of unknown words, has a modest performance in guessing the other morphological attributes of unknown words. In fact, the lexical features used for unknown words

POS tags, their suffixes and prefixes, are not appropriate for guessing some of their other morphological attributes. For example, the Arabic nouns are determined or made definite by prefixing the definite article /al-/; by using the noun as first term of an iDaafa (annexation structure); or by suffixing a possessive pronoun to the noun. In Arabic writing, the definite article and possessive pronouns are always attached to the stem. However, because we are using segmented version of the Arabic Quranic Corpus (morpheme-based version), these clitic are separated from the stem and represented as standalone units. In this case, we think that utilizing the lexical features of the context (previous and next words) is more appropriate than the lexical features of the word itself. Moreover, the indefinite markers (suffix sound), which corresponds to the use of "a" or "an" in English, are no longer used in most of the modern Arabic writing (and also we used unvocalized version of the Arabic Quranic Corpus).

Table 2- Intermediate results of all the simple (seven) classifiers

Classifiers and Data		Tag set Type & Size		Unknown %	Overall %	Training Time	Testing Time
C1	D1	POS	45	86.13	94.27	0.0489	0.446
C2	D2	Person	4	82.00	96.27	0.0347	0.0402
C3	D3	Gender	3	75.5	96.11	0.0332	0.0450
C4	D4	Definiteness	3	68.8	95.75	0.0324	0.0430
C5	D5	Number	4	61.6	94.73	0.0336	0.0435
C6	D6	Tense	4	82.24	97.88	0.0316	0.0402
C7	D7	Voice	3	85.4	98.132	0.0309	0.0403

Finally, a hierarchical combinations algorithm is used to combine these intermediate results. The hierarchical combination tries to incorporate more linguistic dependencies which have been lost between the classifiers outcomes. However, not all attributes are relevant to all POS tags. For example, nouns have values for 'definition', 'gender', 'number' and 'person' and can only have 'irrelevant' as the possible value for 'tense' and 'voice''. Conjunctions have no attributes with values other than 'irrelevant'. In this method, we first predict the main POS of the target word, and take this prediction to be true. We then combine the outcomes of only a subset of the other classifiers, determined by the main POS. For example, if the output of the POS classifier is noun, we do not have to look into the output of the classifiers which predict the 'tense' and 'voice'' morphological attributes or even to run them. The results of the hierarchical combination are shown in Table 3. The results, in both Table 1 and Table 3 , show that even we have achieved high single attribute morphological tagging accuracy, the combined results is so low compared to the direct classification. This can be returned knowing that breaking down the morphological disambiguation into many single morphological attribute tagging

problems leads to lose a lot of linguistic dependencies which are exist between these attributes, explains this result.

In general, both methods, direct classification method and single-attribute classifiers combination method, have advantage and disadvantages. The direct classification method is more accurate and it retains the linguistic dependencies between the morphological attributes. However, it causes data sparseness and its performance is so slow. On the other hand, the single-attribute classifiers combination method is very fast and it does not suffer from data sparseness problem. The time complexity of single morphological attribute classifiers combination method is small $O(n\times F_1^3)+O(n\times F_2^3)+....+O(n\times F_7^3)+O(n)\approx O(n\times F_1^3)$, where $O(n\times F_i^3)$ is the time complexity of classifier C_i and $O(n)$ is the combination algorithm's time complexity. However, it is less accurate and it does not preserve the linguistic dependencies between the morphological attributes. From that, it is clear that to design a better way to handle morphological disambiguation, we should balance between the two ways. In the next subsections, we describe N-attribute classifier combination methods, where N=2 and N=3.

Table 3- Results of the single-attribute classifiers combination method

Combination Type	Unknown %	Overall %	Testing Time
Simple Combination	30.17	84.67	0.446
Hierarchical Combination	30.66	85.00	0.49

2.3. Pair-Attributes Classifiers Combination

The main idea behind this decomposition is that POS attribute is the main morphological attribute and the other morphological attributes are POS-specific. The number of related attributes is fixed for each main POS category. In this method, the original training data is decomposed into six (M=6) new training data {D_1, D_2,…, D_6 }. Each one of these training data is tagged with the possible values which resulted from the combination of two (N=2) morphological attributes. The six morphological groups are POS.Person, POS.Number, POS.Gender, POS.Definiteness, POS.Tense and POS.Voice. Then, six classifiers {C_1, C_2,….., C_6} are used. Each classifier C_i is trained with training data D_i. Each classifier predicts a small set of possible values resulted from the combination of two morphological attributes (POS and one of the remaining morphological attributes). The intermediate results of all these classifiers are shown in Table 4.

Table 4- Intermediate results of all the pair-attributes classifiers

Classifier and Data		Tag set Type &	Size	Unknown%	Overall%	Training Time	Testing Time
C1	D1	POS + Person	52	79.81	93.00	0.052	0.661
C2	D2	POS + Gender	65	67.64	92.83	0.0591	1.224
C3	D3	POS + Definiteness	49	78.35	93.41	0.051	0.592
C4	D4	POS+ Number	69	63.99	92.69	0.061	1.425
C5	D5	POS + Tense	47	81.02	93.90	0.0513	0.520
C6	D6	POS+ Voice	46	83.70	93.84	0.0500	0.495

Finally, a majority and hierarchical combination algorithm is used to combine these intermediate results. As shown in Table 4, each classifier predicts for each word a POS tag and a value of additional morphological attribute. The predicted POS tags for a word may differ from classifier to another. In this case, we select POS tag which is predicated by the majority. If there is no majority, we select the POS tag that is predicated by the group which includes the classifier whose POS tagging accuracy is the highest. The POS tagging accuracy of each classifier is shown in Table 5. From these results, we have two important observations. First, the POS tagging accuracy, when tagged with additional attribute is better than the POS tagging accuracy when tagged alone (see the result of classifier C_1 in Table 4). This means using additional attributes helps to disambiguate the main POS tag. Second, the majority combination algorithm gives higher POS tagging result than the best classifier. After the POS tag is chosen, we select the output of the relevant attributes' classifiers. For example, if the majority of the classifiers {C_1,C_2,C_3,C_4 and C_5} choose the "ADJ" as the correct POS tags for the target word , we then look into the outcomes of classifiers C_1(for Person),C_3 (for definiteness)and C_4 (for number) and ignore other classifiers. The results of the majority and hierarchical combination are shown in Table 6. The results show that pair-attributes classifiers combination method improves the overall results (4.2%) over single-attribute classifiers combination method. Compared to the direct classification, the direct classification works slightly better than that pair-attributes classifiers combination method. However, the pair-attributes classifiers combination method is much faster than direct classification.

Table 5- POS overall tagging accuracies of pair-attributes classifiers

Classifiers	Overall POS Tagging Accuracy
C_1	94.58
C_2	94.53
C_3	94.28
C_4	94.58
C_5	94.44
C_6	94.33
Majority	95.00

Table 6- Results of the majority and hierarchical combination method (pair-attributes classifiers)

	Tag set Size	Unknown%	Known%	Overall%	Testing time
Majority + Hierarchical Combination	184	42.34	90.82	89.23	2.05

2.4. Triple-Attributes Classifiers Combination

In this method, the seven morphological attributes is divided into three groups; each group contains three attributes. They are divided according to the following ideas:

a) POS attribute is the main morphological attribute. Others are POS-specific. So, POS attribute is part of each group.

b) Related or co-occurred attributes are grouped together. So, tense and voice attributes are in the same group.

c) Avoid sparseness i.e. attributes with less possible combinations are preferred to be together. So from the four remaining attributes Person(4 values), Gender(3 values), Definiteness(3 values) and Number(4 values), Person and Gender are selected to be in one group and the other two in another group.

According to that, the seven morphological attributes are divided into three sets, N=3: {POS, Person, Gender},{ POS, Definiteness, Number }and { POS , Tense , Voice }. Then, three, M=3, training data {D_1, D_2, D_3} are created. Each one is tagged with the possible values of one of these sets. We use three classifiers, {C_1, C2, C3} where classifier C_i is trained with training data D_i. The intermediate results of the three classifiers on the three training data are shown in Table 7.

Table 7- Intermediate results of all the three-attribute classifiers

Classifier and Data		Tag set Type & size		Unknown	Overall	Training Time	Testing Time
C1	D1	POS + Person + Gender	78	67.4	93.22	0.0672	1.9231
C2	D2	POS +Definiteness + Number	80	60.34	91.92	0.0671	2.0532
C3	D3	POS + tense voice	50	78.59	93.42	0.0506	0.5852

Finally, a combination algorithm is used to combine these intermediate results. In the three-attribute classifiers combination method, each classifier predicts for each word a triple tag (_._._). The first slot is for the POS tag. For the two remaining slots; each one is for a value of one of the two additional attribute. So, we have three POS tags, each one is predicted by one classifier. These three POS tags may be equal or different. If they are different, we choose the POS tag predicted by the classifier (C_1) whose POS tagging accuracy is the highest. C_1 works better than all other classifiers and also works better than the majority voting. So, we do not use the

majority voting combination technique. Instead, we directly use the outcomes of C_1. The POS tagging accuracy of each classifier is shown in Table 8. Moreover, the overall tagging accuracy of each attribute is shown in Table 9. What is important to be noted here is that the POS overall tagging accuracy achieved by C_1 is high (95.20%) compared to most of full statistical models described before, especially C_1, which is trained and tested using data annotated with only POS morphological attributes. This indicates that including some morphological attributes helps clearly to disambiguate the main POS tag. However, after the POS tag is selected; we look of the value of the relevant attributes and ignore others. The results of the combination are shown in Table 10. This result shows that the three (triple)-attribute classifiers combination algorithm achieves comparable accuracy with the direct classification algorithm. In addition, it also outperforms both single-attribute classifiers combination algorithm and pair-attribute classifiers combination algorithm. However, this method is very fast and it does not suffer from data sparseness problem as the single-attributes classifiers combination algorithm and it is accurate as the direct classification algorithm. Moreover, the three-attribute classifiers combination method is much faster than direct classification. The three (triple)-attribute classifiers combination algorithm provides a solution which compromises between the two extremes (the direct classification algorithm and the single-attributes classifiers combination algorithm).

Table 8- Overall Tagging Accuracy of POS Attribute Achieved By the triple-Attributes Classifiers

Classifiers	Overall POS Tagging Accuracy
C_1	95.20
C_2	94.48
C_3	94.44
Majority	95.10

Table 9- Overall tagging accuracy of each attribute achieved by the triple-attribute classifiers

Attribute	The Used Classifier	Overall Accuracy
Person	C_1	97.65
Gender	C_1	97.62
Definiteness	C_2	98.15
Number	C_2	97.00
Tense	C_3	98.72
Voice	C_3	98.78

Table 10 Results of the triple-attribute classifiers combination method

	Tag set Size	Unknown%	Known%	Overall%	Testing Time
Best classifier + Hierarchical Combination	184	60.34	91.79	90.5	2.3

3. Conclusions

Morphological disambiguation of Arabic is a hard task due to its rich inflectional, derivational and templatic morphology. The number of potential morphological tags in Arabic is theoretically so large. In this paper, several stochastic methods to statistical morphological disambiguation for rich morphological languages have been presented. These proposed methods do not depend on manually constructed linguistic knowledge such as a dictionary and morphological rules. First, this paper evaluates the direct statistical disambiguation method and the single-attribute classifiers combination method. Then, new methods based on the combination of several N-attribute Trigram HMM classifiers are proposed. In these methods, the multi-attributes morphological tagging problem are decoupled into several N-attributes tagging sub-problems. Then, several classifiers are used to solve each sub-problem. Finally, the outcomes of all n-attributes classifiers are combined. In this way, we try to reduce the data sparseness problem and to retain the linguistic dependencies between the morphological attributes. Among models that we have developed and tested for morphological disambiguation of Arabic, the three (triple)-attribute classifiers combination algorithm achieves a good accuracy, does not suffer from the data sparseness, and saves run time.

In addition, while our experiments are limited to the Arabic language and to specific number of morphological attributes, the models presented thus far in this paper are language independent in nature and applicable to any number of morphological attributes. We believe, therefore, that the experiments will be applicable to other morphologically complex languages.

We believe that these results can be further improved in various ways. First, by improving the performance of the unknown word handling algorithms for each basic classifier by utilizing different lexical features depend on the morphological attributes which the classifier is used to predict their values. The basic classifiers can also benefit from careful tuning of their parameters. Finally, we also believe that linguistic exploration, based on deeper error analysis, and to develop hard constraints which can be used to reduce the error rate of the combination module.

References

1. Hakkani-Tür, D. Z., Oflazer, K., & Tür, G. (2002). Statistical morphological disambiguation for agglutinative languages. Computers and the Humanities, 36(4), 381-410.
2. Van Halteren, H., Zavrel, J., & Daelemans, W. (2001). Improving accuracy in word class tagging through the combination of machine learning systems.Computational linguistics, 27(2), 199-229.
3. Schmid, H., & Laws, F. (2008, August). Estimation of conditional probabilities with decision trees and an application to fine-grained POS tagging. InProceedings of the

22nd International Conference on Computational Linguistics-Volume 1 (pp. 777-784). Association for Computational Linguistics.

4. G. Orphanos, "Computational Morphosyntactic Analysis of Modern Greek " Ph.D., Department of Computer Engineering and Informatics, University of Patras, 2000.
5. M. Levinger, U. Ornan, and A. Itai, "Morphological disambiguation in Hebrew using a priori probabilities," *Computational Linguistics* vol. 21, pp. 383-404, 1995.
6. Habash, N. Y. (2010). Introduction to Arabic natural language processing.Synthesis Lectures on Human Language Technologies, 3(1), 1-187.
7. Farghaly, A., & Shaalan, K. (2009). Arabic natural language processing: Challenges and solutions. ACM Transactions on Asian Language Information Processing (TALIP), 8(4), 14.
8. Tachbelie, M. Y. (2010). Morphology-based language modeling for Amharic(Doctoral dissertation, Hamburg, Univ., Diss., 2010).
9. Márquez, L. (1999). Part-of-speech Tagging: A Machine Learning Approach based on Decision Trees.
10. Hajič, J., & Hladká, B. (1998, August). Tagging inflective languages: Prediction of morphological categories for a rich, structured tagset. In Proceedings of the 17th international conference on Computational linguistics-Volume 1 (pp. 483-490). Association for Computational Linguistics.
11. Shacham, D. (2007). Morphological disambiguation of Hebrew (Doctoral dissertation, University of Haifa).
12. Shacham, D., & Wintner, S. (2007, June). Morphological Disambiguation of Hebrew: A Case Study in Classifier Combination. In EMNLP-CoNLL (pp. 439-447).
13. Oflazer, K., & Kuruöz, İ. (1994, October). Tagging and morphological disambiguation of Turkish text. In Proceedings of the fourth conference on Applied natural language processing (pp. 144-149). Association for Computational Linguistics.
14. Dukes, K., & Habash, N. (2010, May). Morphological Annotation of Quranic Arabic. In LREC.
15. Habash, N., & Rambow, O. (2005, June). Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (pp. 573-580). Association for Computational Linguistics.
16. Smith, N. A., Smith, D. A., & Tromble, R. W. (2005, October). Context-based morphological disambiguation with random fields. In Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing (pp. 475-482). Association for Computational Linguistics.
17. Dukes, K., Atwell, E., & Sharaf, A. B. M. (2010, May). Syntactic Annotation Guidelines

for the Quranic Arabic Dependency Treebank. In LREC.

18. Albared, M., Omar, N., & Ab Aziz, M. J. (2011). Developing a competitive HMM Arabic POS tagger using small training corpora. In Intelligent Information and Database Systems (pp. 288-296). Springer Berlin Heidelberg.
19. Albared, M., Omar, N., & Ab Aziz, M. J. (2011). Improving arabic part-of-speech tagging through morphological analysis. In Intelligent Information and Database Systems (pp. 317-326). Springer Berlin Heidelberg.
20. Thede, S. M. (1999). Parsing and tagging sentences containing lexically ambiguous and unknown tokens (Doctoral dissertation, Purdue University).