Journal of Theoretical and Applied Information Technology
31st August 2016. Vol.90. No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

PROBABILISTIC ARABIC PART OF SPEECH TAGGER WITH UNKNOWN WORDS HANDLING

[1]Mohammed Albared, [2]Tareq Al-Moslmi, [3]Nazlia Omar, [4]Adel Al-Shabi, [5]Fadl Mutaher Ba-Alwi
[2,3,4] Centre for Artificial Intelligence Technology, Faculty of Information Science and Technology,
Universiti Ke-bangsaan Malaysia, 43600 Bangi, Selangor, Malaysia
[1,5]Faculty of Computer and Information Technology, Sana'a University, Yemen
E-mail: [1]mohammed_albared@yahoo.com , [2]tareq.almoslmi@gmail.com, [3]nazlia@ukm.edu.my , [4]adel.alshabi@gmail.com, [5]dr.fadlbaalwi@gmail.com

ABSTRACT

Part Of Speech (POS) tagger is an essential preprocessing step in many natural language applications. In this paper, we investigate the best configuration of trigram Hidden Markov Model (HMM) Arabic POS tagger when small tagged corpus is available. With small training data, unknown word POS guessing is the main problem. This problem becomes more serious in languages which have huge size of vocabulary and rich and complex morphology like Arabic. In order to handle this problem in Arabic POS tagger, we have studied the effect of integrating a lexicon based morphological analyzer to improve the performance of the tagger. Moreover, in this work, several lexical models have been empirically defined, implemented and evaluated. These models are based essentially on the internal structure and the formation process of Arabic words. Furthermore, several combinations of these models have been presented. The POS tagger has been trained with a training corpus of 29300 words and it uses a tagset of 24 different POS tags. Our system achieves state-of-the-art overall accuracy in Arabic part of speech tagging and outperforms other Arabic taggers in unknown word POS tagging accuracy.

Keywords: *Part of Speech Tagger, Arabic Language, Unknown Word Guessing.*

1. INTRODUCTION

Part of speech disambiguation is the ability to computationally determine which part of speech of a word is activated by its use in a particular context. Automatic text tagging is an important preprocessing step in many NLP applications such as information extraction, question answering and machine translation. POS tagging is a nontrivial problem. It cannot exclusively consist of a lexicon due to the MorphoSyntactic ambiguity, and the existence of unknown words, that is, words that have not been previously seen in the annotated training set. Unknown words are major problem in any tagging systems, and always decrease the performance of the systems. The accuracy of part-of-speech (POS) tagging for unknown words is significantly lower than that for known words.

The processing of unknown words is so important due to several reasons. First, the unknown words play an important role in the meaning of a sentence more than known words; unknown words are specialized words and hold more semantic information than known word [1]. This is because most of the unknown words belong to open POS classes such as nouns and verbs and unlikely to be in the closed classes such as particles. Second, the performance of the POS tagger in unknown word tagging is a measure of its robustness and reliability, which is, its ability to tag document from different domains or language varieties without substantially decrease on its performance. Finally, the improvement of unknown words tagging contribute to the overall accuracy of the POS tagger. For these reasons, properly POS tagging of unknown words is so important, so the information carried by them can be used correctly in future steps of an NLP system.

Unknown word POS tagging is a substantial problem in Arabic POS tagging due to several reasons. First, the lack of large and free publicity available annotated corpora. Second, Arabic language is one of the richest languages in term of vocabulary [2], In the DIINAR.1 resource, the effective number of simple word forms is 7,774,938 [3]. As a result, to design a reliable and robust statistical POS tagger, we need extremely large annotated corpus. Third, Arabic language is inflected language with rich and complex morphology. Finally, the orthographic ambiguity; the form of certain letters in Arabic script allows

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

suboptimal orthographic variants of the same word to coexist in the same text.

In this work, we employ the well-known trigram HMM POS tagging architecture for tagging Arabic text – our baseline tagger implementation is influenced by Brants et.al. [4]. During the implementation of our baseline tagger, we observed that the suffix guessing does not performed well on Arabic unknown words. This is due to the limitation in the training data size and language characteristics. So, to cope with the unknown words problem first, we study how information supplied by a lexicon based Morphological Analyzer (MA) can be used to improve the accuracy of the system. Then, we define, implement and evaluate several lexical models based on the internal structure of Arabic word i.e. the word formation process. Experimental results show that the proposed approaches achieve very encouraging results, although the training is performed on very small size corpus.

The rest of the paper is organized as follows. Sec. 2 discusses related works. Sec. 3 and Sec. 4 describes our tagset and corpora. Sec. 5 gives necessary details about Arabic word formation. Sec. 6 describes our baseline HMM tagger. In Sec. 7 we discuss the modifications to better handle unknown words POS tagging in Arabic text. Sec. 8 gives Experimental results. Finally, conclusions and future work appear in Sec. 9.

2. RELATED WORK

Research on POS tagging has a long history. Numerous approaches have been successfully applied to POS tagging. The POS tagging techniques in the literature can be classified into the following:

POS tagging techniques in the literature can be classified into the following:

- Rule-based part-of-speech tagging which is based on a lexicon and a set of disambiguation rules.
- Supervised POS tagging: these approaches use machine-learning techniques to learn a classifier from labeled training sets such as Maximum Entropy Model [5] , Hidden Markov Model [4], [6],Conditional Random field [7], Cyclic Dependency Networks [8] and Support Vector Machine [9], [10].
- Unsupervised POS tagging: these approaches do not require pre-tagged training data, but rely on dictionary information.

Previous work on POS tagging has utilized different kind of features to tackle unknown word POS tagging. These features are mainly based on word substring information, word context information and/or global information. Weischedel et al. [11] create a probability distribution for an unknown word based on certain features: word endings, hyphenation, and capitalization. Brill et.al. [12] uses suffix information with transformation rules. Ratnaparkhi et al. [5] uses character n-gram prefixes and suffixes, and spelling cues such as capitalization, hyphens, and numbers. Brants et.al. [4] uses the linear interpolation of fixed length suffix model for unknown word handling. Nakagawa et.al. [13] uses global information and local information. They model the probability distribution of the POS of all the occurrences of unknown words with the same lexical form in a document. The parameters were estimated using Gibbs sampling. Agic et al. [14] and showed that the performance of high inflected language POS tagger can be improved significantly by integrating the output of morphological analyzer.

Recently, several works have been proposed to Arabic POS tagging such as [15]–[21], for more details about Arabic works in POS tagging see Albared et al. [22]. Among all these works, AlGahtani et al. [16] and Marsi et al. [18] reported their taggers performances on unknown word POS tagging which are 67.0% and (80 %-85%) respectively. However, the reported results still less than achieved results in other languages like English. Marsi et al. [18] used prefix, suffix, two previous words tags and one next word tag to handle unknown words. In addition, Al Shamsi et al. [17] and El Hadj et al. [15] used HMM for Arabic POS tagging. Both of them used 1000 words as test set. But, they worked under closed vocabulary assumption.

3. THE TAGSET FOR ARABIC POS TAGGING

Our tagset have been inspired by Arabic TreeBank (ATB) POS Guidelines [23]. The used tagset consists of 24 tags (see table 1). This tagset is a refinement of the Arabic TreeBank tagset, which is consist of 23 tags, used by Mansour et al [19], Diab et al [20] and Habash et al [21]. We only add some modifications to handle some linguistic limitation on previous Arabic taggers. The first one, we introduce a tag for the Broken Plural (BP) to distinguish between it and the singular noun. Unlike English irregular plural, which is uncommon, Arabic broken plural is very common. BPs form 40% of the plurals and the remaining

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

percentage 60% is for the other types of plurals: sound masculine and feminine plurals [24]. In our annotated corpus, BPs form 55% of the plurals. Moreover, BPs constitute 10% of any Arabic text [25]. Several works in Arabic NLP have been proposed to identify BP in Arabic text [25]–[27]. However, previous Arabic taggers do not identify BP as independent tag. Most of the time, BPs are tagged as singular noun which leads to lose a lot of information such as Mansour et al [19], Diab et al [20] and Habash et al [21]. The main word formation process in Arabic languages is inherently non-concatenative; the BP is the best example of this non-concatenative morphology [27]. We can measure the performance of our algorithms on handling non-concatenative unknown words by measuring its performance on handling unknown words which are BP. The second modification, our tagset does not include NO_FUNC (no solution chosen) tag, which is used as a tag in the above mentioned Arabic TreeBank tagset. They use this tag for any Arabic word with no selected solution [28]. Finally, we distinguish between inflected and non-inflected verbs.

4. THE TRAINING CORPUS

Our corpus consists of 29340 manually annotated word forms from two types of Arabic texts. Over 17000 word forms come from old Arabic text or what is called "Traditional Arabic text" and another 12000 are coming from modern standard Arabic. The main difference between the two types of text is only Out Of Vocabulary words. A few old Arabic words are rarely used nowadays writing. In contrast, some new technical terms and new words have entered common usage. We use this corpus to train and test our tagger. We spilt the corpus into training set with size 22800 words and test set with size 6540 words.

5. ARABIC WORD STRUCTURE

Arabic word form is either simple or complex (see Figure 1). The simple form of Arabic word consists of prefix, stem and suffix. The complex form consists of proclitics, the simple form and enclitics. Clitics (proclitics and enclitics) have their own POS tags. Tagging at complex word form level increase the data sparseness problem (increase unknown word problem) and increase the complexity of the tagset [28][29]. Furthermore, Barhaim et al. [29] showed that POS tagging using simple word form outperforms tagging using complex word form in Semitic languages. However, throughout this research the simple word form will be termed "word". We assume the segmentation as a preprocessing step of the POS tagger.

Arabic words are quite different from English words, and the word formation process for Arabic words is quite complex. The main formation of English word is concatenative. In contrast, the main word formation process in Arabic languages is inherently non-concatenative [30].

The word in Arabic language can be described as combinations of two morphemes: a root and pattern. The root is a sequence of three (rarely two or four) characters which is called radicals. The pattern is a combination of augmented characters "احرف الزيادة" (vowel characters and it can be consonants), with generic (or variables) characters into which the Root Radical Characters (RRC) are being inserted (throughout this works, we use the English letter X to represent the pattern generic characters). The augmented characters (sometimes called fixed characters) are fixed in each pattern. Words are derived by interdigitating roots into patterns: the first radical is inserted into the first generic character, the second radical fills the second generic and the third fills the last generic as shown in Table 2. Arabic has a small number, a few hundreds, of patterns and a few thousand of roots.

The Arabic alphabet has 28 basic letters. Arabic word letters are divided into two sets. The first one is the root radical characters. Any Arabic letter can be root radical character. Root radical characters in Arabic word do not play any role in the detection of the word possible POS tags. For example, in the Arabic word.

"متخاصمون", the three characters خ,ص and م are root radical characters. However, we can replace them by other three Arabic characters such as (ص, ق ,د) to produce other Arabic word "متصادقون" which have different meaning but both words are SPN. The second set is the augmented characters. Each augmented character can be only one of the these Arabic ten characters {ا,ه,ي,ن,و,م,ت,ل,أ,س}. However, the augmented characters associated with their position in the word may play a critical role in determining the possible POS tag of the word for example Arabic words "يتصاعد","يتفاهم","يتصالح","يتعامل" all have three augmented characters 'ي','ت'and 'ا' in the first ,second and the fourth position and all these words are either PSV or VBP .

Table 1. Word Deviation Process Of Some Arabic Words From The Root "كتب" With Different Patterns

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Pattern	Root	The resulting word
XXX	كتب	كتب
XX!X	كتب	كتاب
مXXوX	كتب	مكتوب
مXXXة	كتب	مكتبة
X!XX	كتب	كتاب

6. OUR BASELINE MODEL :THE HMM POS TAGGER

Hidden Markov Model (HMM) is a well-known probabilistic model, which can predict the tag of the current word given the tags of one previous word (bi-gram) or two previous words (trigram). The HMM tagger assign a probability value to each pair $< w_1^n, t_1^n >$, where $w_1^n = w_1 ..... w_n$ is the input sentence and $t_1^n = t_1 ....t_n$ is the POS tag sequence. In HMM, the POS problem can be defined as the finding the best tag sequence t_1^n given the word sequence w_1^n. The label sequence t_1^n generated by the model is the one which has highest probability among all the possible label sequences for the input word sequence. This is can be formally expressed as:

$$t_1^n = \arg\max_{t_1^n} \prod_{i=1}^{n} p(t_i | t_{i-1} \ldots t_1).p(w_i | t_i \ldots t_1) \quad (1)$$

$$t_1^n = \arg\max_{t_1^n} \prod_{i=1}^{n} p(t_i | t_{i-1} \ldots t_1).p(w_i | t_i) \quad (2)$$

The first parameter $p(w_i | t_i \ldots t_1)$ is a known as the emission probability and second parameter $p(t_i | t_{i-1} \ldots t_1)$ is known as the transition probability. These two model parameters are estimated from annotated corpus by Maximum Likelihood Estimation (MLE), which is derived from the relative frequencies. Given these two probabilities, we can find the most likely tag sequence for a given word sequence. Using the Viterbi algorithm [31], we selected the path whose overall probability was the highest, and then took the tag predictions from that path. However, MLE is a bad estimator for statistical inference especially, in NLP application, because data tends to be sparse. This is even for corpus with large number of words. Sparseness means that various words are either infrequent or unseen. This leads to zero probabilities being assigned to unseen events, causing the probability of the whole sequence to be set to zero when multiplying probabilities. There are many different smoothing algorithms in the literature to handle the sparseness problem [32], all of them consisting of decreasing the probability assigned to the known event and distributing the remaining mass among the unknown events. In our work, we use linear interpolation of unigram, bigram and trigram maximum likelihood estimates in order to estimate the trigram transition probability:

$$P(t_3 | t_2, t_1) = \lambda_1 P(t_3) + \lambda_2 P(t_3 | t_2) + \lambda_2 P(t_3 | t_2, t_1) \quad (3)$$

where $\lambda_1 + \lambda_2 + \lambda_3 = 1$, so P represents a valid probability distribution. The λs are estimated by deleted interpolation.

For unknown word, we use the linear interpolation of fixed length suffix model for unknown word handling. The probability distribution for a unknown word suffix is generated from all words in the training set that have the same suffix up to some predefined maximum length. Probabilities are smoothed by successive abstraction. This method was proposed by Samuelsson et.al. [33] and implemented for English and German [4].

$$P(t | c_{n-i+1}, \ldots, c_n) = \frac{P(t, c_{n-i+1}, \ldots, c_n) + \theta P(t, c_{n-i+2}, \ldots, c_n)}{1 + \theta} \quad (4)$$

$$\theta = \frac{1}{S-1} \sum_{J=1}^{S} (P(t_j) - \overline{P})^2$$

$$and \quad \overline{P} = \frac{1}{s} \sum_{J=0}^{S} P(t_j)$$

Where $c_{n-i+1}, \ldots, c_n$ represent the last n characters of the word of the words. In addition to word suffix, the experiments utilize the following features: the presence of non-alphabetic characters and the existence of foreign characters. In addition to the suffix guessing model, we define another basic model based on both unknown word prefix and suffix. The main linguistic motivation behind combining affixes information is that in Arabic word sometimes an affix requires or forbids the existence of another affix [34]. Prefix and suffix indicate substrings that come at the beginning and end of a word respectively, and are not necessarily morphologically meaningful. In this model, the lexical probabilities are estimated as follows:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Given an unknown word w, the lexical probabilities $P(\text{suffix(w)}\,|\,t)$ are estimated using the suffix tries as in Equation 4.

Then, the lexical probabilities $P(\text{prefix(w)}\,|\,t)$ are estimated using the prefix tries as in Equation 4. Here, the probability distribution for an unknown word prefix is generated from all words in the training set that have the same prefix up to some predefined maximum length.

Finally, we use the linear interpolation of both the lexical probabilities obtained from both word suffix and prefix to calculate the lexical probability of the word w as in the following equation:

$$P(w|t) = \lambda\, P(\text{suffix}(w)|t) + (1-\lambda)P(\text{prefix(w)}\,|\,t) \quad (5)$$

Where λ is an interpolation factor, experimentally set to X. prefix(w) and $\text{suffix}(w)$ are the first m and the last n characters, respectively. Table 3 summarizes the results of experiments with prefix, suffix and prefix + suffix basic models. The first model (LM1) is TnT suffix guessing algorithm. The second model (LM2) is prefix guessing algorithm. The third model (LM3) is the linear interpolation of both prefix guessing algorithm and suffix guessing algorithm for unknown words. LM3, which combine information from both suffix and prefix, gives a considerable rise in accuracy compared to the suffix guessing method. However, the performances of LM1, LM2 and LM3 in unknown words still far away from what are achieved in other languages. The results also show that some techniques which proved to be effective for some languages does not work well for Arabic languages such as LM1 (suffix guessing algorithm) which proved to be a good indicator for unknown word POS guessing in English and German [4]. In the next section, we discuss our effort to improve the accuracy of the unknown word predictor. We combine the weighted output of MA with word suffix and prefix information and with word pattern suffix and prefix.

Table 2. The average POS tagging accuracy using the HMM tagger with the basic lexical models.

Model	% of unknown word	Unknown acc.	The overall acc.
LM1(TnT)Suffix guessing algorithm	10.7	66.3	94.7
LM2 Prefix guessing algorithm	10.7	56.4	93.6
LM3 Prefix +suffix guessing algorithm	10.7	69.5	95.0

7. SYSTEM IMPROVEMENT

7.1 Integration of Morphological Information

In order to further improve the tagging accuracy, we integrate morphological information with lexical models. The main reason of our choice of using external MA is based on the fact that suffix tries and successive abstraction algorithm does work well with Arabic language. In our opinion, the main reasons that make this algorithm unsuitable for Arabic language are: 1) data sparseness 2) suffix ambiguity 3) the non-Concatenative nature of Arabic word. A MA is a function that inputs a word w and outputs the set of all its possible POS tags. Note that the size of tags produced by the MA is much smaller than the size of the tagset. Thus, we have a restricted choice of tags as well as tag sequences for a given sentence. Since the correct tag t for w is always in the MA output tags (assuming here that the MA is complete), it is always possible to find out the correct tag sequence for a sentence even after applying the morphological restriction. Since the MA does not assign probabilities to the tags, we address this problem by assuming uniform distribution of the tags proposed by the MA given the word. In our system, we utilize the LDC-distributed Buckwalter Morphological Analyzer for Arabic (BAMA).The BAMA system is based on three tables: prefixes table, stem table and suffixes table. The stem table of BAMA has a very high coverage. Due to the differences between BAMA tagset and our tagset, we implement a mapping function that map each tag produced by BAMA to one tag or more in our used tagset. The new combined models (LM4 and LM5) follow a simple method for using the information from the MA:

1) If the word is in the training corpus, the lexical probabilities are estimated using MLE just as in the basic models, otherwise.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2) If the word is not in the training corpus and it is known to the MA, the MA output the set of possible tags. the lexical probabilities are estimated as follow:
   a. The tags probability distribution is calculated by using appropriate weighting function such as assuming uniform distribution of the tags proposed by the MA.
   b. Then, lexical probabilities are calculated using Bayesian inversion.
   c. Finally, we combine these lexical probabilities with lexical probabilities provided by LM1 or LM3.
3) If the word is also unknown to the MA, the lexical probabilities provided by LM1 or LM3 are used.

However, for fixed input text the number of unknown words can be reduced by enriching the lexicon of the MA. This is a suitable if the tagger is domain specific. But as a general solution for general multi-domain texts, the tagger must be equipped with some models that handle unknown word efficiently without extending the size of the MA lexicon. Moreover, BAMA has many weaknesses in its coverage and in its analysis [28], [34], [35]. Mansour et al. [19] stated that more than 5% of the words in ATB2 and ATB3 cannot be tagged correctly using BAMA unless further data are added to those provided by the morphological analyzer. However, BAMA analyze Arabic words in concatenative manner. It has problem to analyze words with non-concatenative morphology such as broken plurals [36]. This means it is unable to handle unknown words which are non-concatenative. Our objective is to provide a solution to unknown word POS guessing problem which overcome the limitation of the MA and also overcome the need of huge amount of annotated data. In the next section, we will define lexical models which depend on some specific features of Arabic words. These models have the ability to extract the useful information from Arabic words, which are formed either using concatenative morphology or non-concatenative morphology. Then, they use this information to predict the word appropriate POS tag.

7.2 Using Words Internal Structure

Arabic words are quite different from English words. The word formation process for Arabic words is quite complex. The main formation of English word is concatenative i.e. simply attaching affixes to the beginning and the end of the stem. Hence, the word suffixes are strong indicator for the word POS class. Brants et.al. [4], for example, showed that an English word ending in the suffix -able is very likely to be an adjective. In contrast, the main word formation process in Arabic languages is inherently non-concatenative [30]. Thus, Arabic word (minimal word form) suffixes are ambiguous, short and sparse. For example, most of the time Arabic words, which are derived from the same root, share the same suffix even if they have different POS. Moreover, words, which belong to the same POS class, often have different suffixes (see Table 4).

As we state in Sec. 5, Arabic words is derived by inserting root radical characters into pattern's generic characters. Arabic words characters are divided into root radical characters or augmented characters. While ten characters {ا,ه,ي,ن,و,م,ت,ل,أ,س},which is called Augmented Characters(AC), of the Arabic 28 can be used as root radical or augmented characters , any character of the remaining 18 characters can be used only as root radical characters. The augmented characters appear in Arabic words and their patterns so they are sometimes are called fixed characters [2]. In contrast, root radical characters only appear in the Arabic words and they are replaced with generic characters (or variables) in its pattern. However, the reverse process to word derivation is the pattern identification (or root extraction). The pattern identification is the process that identifies the root radical characters in an Arabic word and replaces them with generic characters.

Table 3. List Of Some Arabic Words Derived From Roots "عمل"And "صنع" And Their POS. The Table Shows The Ambiguity And The Sparseness Of Arabic Word Suffixes

Arabic words	Pattern	Arabic pattern	POS
عمل|صنع	XXX	فعل	PV
سنعمل| سنصنع	سXXX	سنفعل	VBS
عمل|صنع	XXX	فعل	SN
معمل|مصنع	مXXX	مفعل	SN
معامل|مصانع	مXX|X	مفاعل	BP
عمال|صناع	X|XX	فعال	BP
يعملون|يصنعون	يXXXون	يفعلون	VBS
متعاملون|مصنعون	مXXXون|متX|XXون	متفاعلون|مفعلون	SNP

The pattern of Arabic word is a good indicator of its possible POS tags. In addition, patterns can be used to overcome the need of huge annotated data to cover the language vocabularies. All Arabic words which belong to open classes can be mapped to few thousand of patterns. Furthermore, by

Journal of Theoretical and Applied Information Technology
31st August 2016. Vol.90. No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

removing root radical characters from Arabic words, suffixes become less ambiguous, less sparse and long (see Table 5). But, it's not that easy to fully utilized patterns information. Pattern identification (or root extraction) in itself is a complicated task in Arabic NLP. In our current work, we try to balance between the benefit that we can get from the pattern and the complication of the pattern identification. We propose a light pattern identification algorithm to map the Arabic word to its pattern. The algorithm works as follow: given an Arabic word which belong to open class: first, check the words if it contains one character or more from radical characters only set, replace them with generic character "X". Second, for the remaining characters, we use some positions rules, which proposed by (Sonbol et al., 2008), to detect if they are root radicals or augmented characters. We called the pattern produced by this algorithm "Augmented Character Form" (ACF). We use this algorithm to map each non functional word in the dictionary obtained from the training corpus to its ACF. Then, we estimate the emission probability (the lexical model) for each unique ACF.

We use augmented letter tree, to represent the lexical model. Finally, for each unknown word in the test set, we estimated the probability of its ACF's suffix using suffix tries and successive abstraction. Then, we combine this probability with the output of the MA. The algorithm is described more formally as follow (step 1 and 2 are performed in the training phase where step 3 is in the test phase):

1) First, for each word W in the dictionary obtained from the training data : if one of its possible tags belong to the open classes then convert it to its augmented character form(ACF) as follow:
   For each character C in W:
   If C $\notin$ AC, then, we replace it with the generic character "X ".
2) Else if C $\in$ AC then we checks if c is augmented or root radical character using the position rules and if its root radical character, we replace it with the generic character "X ".
3) If two or more words have the same ACF, we represent them as one entry (ACFj). The possible tags of the resulting ACF is equal the possible tags of all of its words. The probability distribution of ACFj given a tag t is calculated as in the following equation :

$$P\left(ACF_j \mid t\right) = \sum_{i=1}^{n} P(w_i \mid t) \qquad (6)$$

Where $w_1,\ldots, w_n$ are words that have the same ACF_j.

4) Finally , For each unknown word in the test set, we do the following:
   a) The word is converted to its ACF.
   b) The lexical probabilities $P(\text{ACF\_suffix} \mid t)$ and $P(\text{ACF\_prefix} \mid t)$ are estimated using the suffix tries and prefix tries as in Sec. 5. The only difference that we replace the word by its ACF.
   c) We combined this information with the MA output, if the word is known to the MA.

8. EXPERIMENTS AND EVALUATION

The main purpose of this work is to study the behavior of different lexical models for HMM POS tagger, in order to determine the best way to handle unknown words POS guessing for Arabic language especially, when small amount of data is available. We evaluate these lexical models on the test set.

We have a total of six models. The same training data has been used to estimate the parameters for all the models. Moreover, the same test set has been used to evaluate all the models. The size of the test set is 6540 words in which 700 words are unknown. We define the tagging accuracy as the ratio of the correctly tagged words to the total number of words. Results are summarized in Table 6.

Table 4. The Average Tagging Accuracy Of Arabic Text Using The Improved Lexical Models

Model	% of unknown word	Unknown acc.	The overall acc.
(LM3)word Prefix+ suffix guessing	10.7	69.5	95.0
(LM4) MA+ suffix	10.7	83.7	96.6
(LM5) MA+ word suffix+ word prefix	10.7	83.5	96.6
(LM6) MA+ACF suffix +ACF prefix	10.7	88.3	97.1

We find that in both HMM based models (LM4 and LM5), the use of a morphological analyzer with word affixes improve the accuracy with respect to the basic models (see Table 3 and Table 6). A significant increase in the unknown word POS tagging accuracy and consequently in the overall accuracy is clearly noticeable. As we have noted already the use of MA and word affixes

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

improve the accuracy of the POS tagger. But what is significant to note is that the percentage of improvement is higher when we use of MA and ACF affixes. The results of the experiments using LM6, which combines information from word morphological information and ACF suffix and prefix, show a considerable increase over all other approaches.

In addition, Table 7 compares our works in Arabic unknown words POS guessing with all related Arabic works. Our combined model (LM6) outperforms all related works on Arabic POS tagging that tackle unknown word problems, although our training data is small.

Table 5. Comparison Of Our Results In Unknown Word POS Tagging With Other Related Arabic Taggers

Tagger	The Main technique	% of unknown word in the test set	Size of unknown words	Unknown acc.
Marsi 2005	MB	%6.6	947	%73
AlGahtani 2009	TBL	%5.3	790	%85
LM6	HMM	%10.6	700	%88.3

9. CONCLUSION AND FUTURE WORK

Unknown words tagging is a serious problem in POS tagging especially when small annotated data is available. The impact of this problem increases in languages which have huge vocabulary and rich morphological system like Arabic.

In this paper, we have investigated the best configuration of second order HMM POS tagger for Arabic when the training corpus is small. We have proposed several lexical models based on internal specific features of Arabic words. In addition, external morphological analyzer has been integrated with the POS tagger to improve the tagger results. Furthermore, we have presented several combinations of these lexical models. The best result is achieved by the combined lexical model which combines the weighted output of the morphological analyzer and affixation tries of word augmented character form (pattern form).Our tagger achieves the state of art in Arabic text tagging and outperforms other Arabic taggers in unknown word tagging.

Our future direction is to improve the pattern based unknown word predictor. This improvement can be done through several steps. First, we intend to increase the size of training corpus from small sized to medium sized to cover most of the Arabic words patterns. The second step is to improve the pattern identification algorithm so that each unknown word can be mapped to a pattern of known word. Another future direction is to develop new test set to re-evaluate the performance of the tagger. The new test set will include annotated data from multiple domains.

APPENDIX

The Arabic POS tagset used in annotating our corpus has been attached in table 1. In Figure 1, the simple and complex forms of Arabic word "فلمعتقدهم" with one of its possible tags sequence (composite tag) has been explained. Tables 5 show the list of some patterns with their possible POS tags.

REFRENCES:

[1] D. Vadas and J. R. Curran, "Tagging unknown words with raw text features," in Proceedings of the Australasian Language Technology Workshop, 2005, pp. 32–39.

[2] M. A. M. E. Ahmed, "ALarge-SCALE COMPUTATIONAL PROCESSOR OF THE ARABIC MORPHOLOGY, AND APPLICATIONS," Faculty of Engineering, Cairo University Giza, Egypt, 2000.

[3] R. Abbès, J. Dichy, and M. Hassoun, "The Architecture of a Standard Arabic lexical database: some figures, ratios and categories from the DIINAR. 1 source program," in Proceedings of the Workshop on Computational Approaches to Arabic Script-based Languages, 2004, pp. 15–22.

[4] T. Brants, "TnT: a statistical part-of-speech tagger," in Proceedings of the sixth conference on Applied natural language processing, 2000, pp. 224–231.

[5] A. Ratnaparkhi and others, "A maximum entropy model for part-of-speech tagging," in Proceedings of the conference on empirical methods in natural language processing, 1996, vol. 1, pp. 133–142.

[6] S. M. Thede and M. P. Harper, "A second-order hidden Markov model for part-of-speech tagging," in Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics, 1999, pp. 175–182.

[7] J. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," 2001.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[8] K. Toutanova, D. Klein, C. D. Manning, and Y. Singer, "Feature-rich part-of-speech tagging with a cyclic dependency network," in Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1, 2003, pp. 173–180.

[9] L. Màrquez and J. Giménez, "A general pos tagger generator based on support vector machines," J. Mach. Learn. Res., 2004.

[10] M. Poel, L. Stegeman, and R. op Den Akker, "A support vector machine approach to dutch part-of-speech tagging," in Advances in intelligent data analysis VII, Springer, 2007, pp. 274–283.

[11] R. Weischedel, R. Schwartz, J. Palmucci, M. Meteer, and L. Ramshaw, "Coping with ambiguity and unknown words through probabilistic models," Comput. Linguist., vol. 19, no. 2, pp. 361–382, 1993.

[12] E. Brill, "Some advances in transformation-based part of speech tagging," arXiv Prepr. C., 1994.

[13] T. Nakagawa, "Multilingual word segmentation and part-of-speech tagging: a machine learning approach incorporating diverse features," Nara Institute of Science and Technology, Japan, 2006.

[14] Ž. Agic and Z. Dovedan, "Improving part-of-speech tagging accuracy for Croatian by morphological analysis," Informatica, vol. 32, no. 4, 2008.

[15] Y. El Hadj, I. Al-Sughayeir, and A. Al-Ansari, "Arabic part-of-speech tagging using the sentence structure," in Proceedings of the Second International Conference on Arabic Language Resources and Tools, Cairo, Egypt, 2009.

[16] S. AlGahtani, W. Black, and J. McNaught, "Arabic part-of-speech tagging using transformation-based learning," in Proceedings of the Second International Conference on Arabic Language Resources and Tools, Cairo, Egypt, 2009.

[17] F. Al Shamsi and A. Guessoum, "A hidden Markov model-based POS tagger for Arabic," in Proceeding of the 8th International Conference on the Statistical Analysis of Textual Data, France, 2006, pp. 31–42.

[18] E. Marsi, A. Van Den Bosch, and A. Soudi, "Memory-based morphological analysis generation and part-of-speech tagging of Arabic," in Proceedings of the ACL workshop on computational approaches to semitic languages, 2005, pp. 1–8.

[19] S. Mansour, K. Sima'an, and Y. Winter, "Smoothing a lexicon-based POS tagger for Arabic and Hebrew," in Proceedings of the 2007 Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources, 2007, pp. 97–103.

[20] M. Diab, K. Hacioglu, and D. Jurafsky, "Automatic tagging of Arabic text: From raw text to base phrase chunks," in Proceedings of HLT-NAACL 2004: Short Papers, 2004, pp. 149–152.

[21] N. Habash and O. Rambow, "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, 2005, pp. 573–580.

[22] M. Albared, N. Omar, and M. J. Ab Aziz, "Arabic part of speech disambiguation: A survey," Int. Rev. Comput. Softw., pp. 517–532, 2009.

[23] M. Maamouri, A. Bies, and S. Kulick, "Enhancing the Arabic Treebank: a Collaborative Effort toward New Annotation Guidelines.," in LREC, 2008.

[24] S. Boudelaa and M. G. Gaskell, "A re-examination of the default system for Arabic plurals," Lang. Cogn. Process., vol. 17, no. 3, pp. 321–343, 2002.

[25] A. Goweder and A. De Roeck, "Assessment of a significant Arabic corpus," in Arabic NLP Workshop at ACL/EACL, 2001.

[26] N. K. A. Alajmi, S. Bin Deris, and S. Alnajem, "Computational Approach to Arabic Broken Derived Nouns Morphology," in Advanced Computer Theory and Engineering, 2008. ICACTE'08. International Conference on, 2008, pp. 704–708.

[27] A. Clark, "Supervised and unsupervised learning of Arabic morphology," in Arabic Computational Morphology, Springer, 2007, pp. 181–200.

[28] S. Mansour, "Combining character and morpheme based models for part-of-speech tagging of Semitic languages," Technion-Israel Institute of Technology, Faculty of Computer Science, 2008.

[29] R. Bar-Haim, K. Sima'An, and Y. Winter, "Part-of-speech tagging of Modern Hebrew

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

text," Nat. Lang. Eng., vol. 14, no. 02, pp. 223–251, 2008.

[30] Y. Cohen-Sygal, "Computational implementation of non-concatenative morphology," University of Haifa, 2004.

[31] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," Inf. Theory, IEEE Trans., vol. 13, no. 2, pp. 260–269, 1967.

[32] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," Comput. Speech Lang., vol. 13, no. 4, pp. 359–393, 1999.

[33] C. Samuelsson, "Handling sparse data by successive abstraction," in Proceedings of the 16th conference on Computational linguistics-Volume 2, 1996, pp. 895–900.

[34] M. A. Attia, "Handling Arabic morphological and syntactic ambiguity within the LFG framework with a view to machine translation," University of Manchester, 2008.

[35] M. Sawalha and E. S. Atwell, "Comparative evaluation of arabic language morphological analysers and stemmers," in Proceedings of COLING 2008 22nd International Conference on Comptational Linguistics (Poster Volume)), 2008, pp. 107–110.

[36] S. Alansary, M. Nagi, and N. Adly, "Towards analyzing the international corpus of Arabic (ICA): Progress of morphological stage," in 8th International Conference on Language Engineering, Egypt, 2008.

Journal of Theoretical and Applied Information Technology
31st August 2016. Vol.90. No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

APPENDIX

Table 6. The Arabic POS Tagset Used In Annotating Our Corpus

Pos Tag	Label	Pos Tag	Label
Conjunction	CC	Broken Plural Noun	BPN
Number	CD	Possessive Pronoun	POSS_PRON
Adverb	ADV	Imperfective Verb	VBP
Particle	PART	Non Inflected Verb	NIV
Imperative Verb	IV	Relative Pronoun	REL_PRON
Foreign Word	FOREIGN	Interjection	INTERJ
Perfect Verb	PV	Interrogative Particle	INTER_PART
Passive Verb	PSSV	Interrogative Adverb	INTER_ADV
Preposition	PREP	Demonstrative Pronoun	DEM_ PROP
Adjective	ADJ	Punctuation	PUNC
Singular Noun	SN	Proper Noun	NOUN_PROP
Sound Plural Noun	SPN	Personal Pronoun	PRON

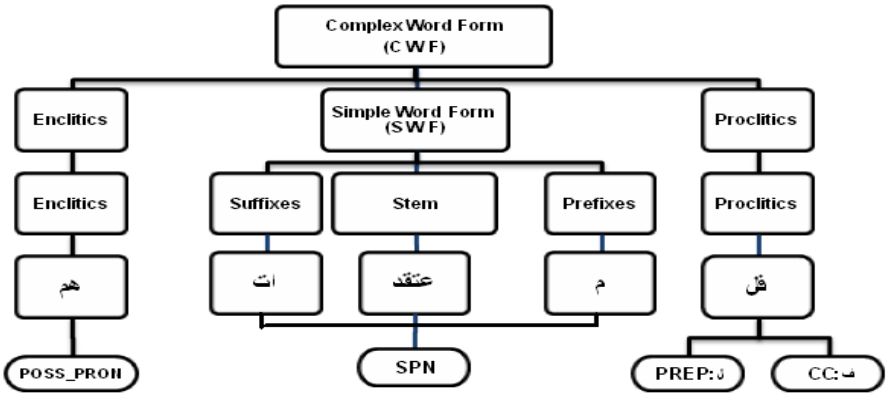

Figure. 1. The Simple And Complex Forms Of Arabic Word "فلمعتقدا/هم" With One Of Its Possible Tags Sequence (Composite Tag) CC+PREP+SPN+POSS_PRON

Table 7. List Of Some Patterns With Their Possible POS Tags

Pattern	Arabic	Examples of Pattern's Words	Pattern Possible Tags
XوXX	فعول	نفوس/صدور/علوم/دخول/غفور/....	BPN,SN,ADJ
اXتXا	افتعال	اجتماع/ اقتصاد / افتتاح/.....	SN
يستXXX	يستفعل	يستخرج/ يستعمل/يستهلك/.....	VBS,PSSV
XXXت	□فعل	□در□/□خرج/□علم/□حدث/.....	IV,PSSV,PV,SN,
يXXXون	يفعلون	يعملون/يصنعون/يسمعون/.....	VBS