International Journal of Engineering & Technology, 7 (3.27) (2018) 125-128

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Part of Speech Tagging for Arabic Long Sentence

Ahmed H. Aliwy[1]*, Duaa A. Al_Raza[2]

[1]*Faculty of Computer Science and Mathematics, Mathematics Department, Iraq.*
[2]*Faculty of Computer Science and Mathematics, Mathematics Department, Iraq.*

Abstract

Part Of Speech (POS) tagging of Arabic words is a difficult and non-travail task it was studied in details for the last twenty years and its performance affects many applications and tasks in area of natural language processing (NLP). The sentence in Arabic language is very long compared with English sentence. This affect tagging process for any approach deals with complete sentence at once as in Hidden Markov Model HMM tagger. In this paper, new approach is suggested for using HMM and n-grams taggers for tagging Arabic words in a long sentence. The suggested approach is very simple and easy to implement. It is implemented on data set of 1000 documents of 526321 tokens annotated manually (containing punctuations). The results shows that the suggested approach has higher accuracy than HMM and n-gram taggers. The F-measures were 0.888, 0.925 and 0.957 for n-grams, HMM and the suggested approach respectively.

1. Introduction

Part of speech tagging, also called word category disambiguation, is choosing the right tag to each word in the sentence from finite set of tags[1][2]. These tags, for example, may be found in a dictionary or a morphological analysis. A word may be has more than one tag from the specific tagset but it has only one suitable tag for the specific sentence. Choosing the correct tag for each word in the sentence is called POS tagging or simply tagging. The POS tagging approaches are developed using linguistic rules, stochastic models and a combination of both [3]. in most cases, a labeled data or corpus is used for training the supervised methods of POS tagging [1]. It can be as classification problem where the tagset as classes and the words as the input.

There are many techniques and approaches used in POS tagging most of them from machine learning. Most of them were applied successfully for Arabic language but with lower accuracy than English language. Rule-based, Hmm, Relaxation labeling, Transformation-Based tagging (Brill), Genetic Algorithms, n-grams Model, Decision trees, Memory based learning, Cyclic Dependency Network, Neural networks tagger, Support Vector Machines, Fuzzy set theory, Boosting, Best match, Maximum Entropy tagger and Combining different taggers are techniques used to tagging system and almost all of them was applied to Arabic language but they need to suitable amount of training corpus. The approaches of Arabic POS tagging are studied with different corpora and tagsets. These studied gave different results because of the nature of Arabic language. The evaluation of POS taggers estimated using precision, recall and f-measure. These measures depend on the nature of the language, size of the tagset, size of the used corpus and other factors.

Arabic language has more complexity, in the level of morphology and syntax, than many other languages. Arabic word can has multiple POSs, for example the word "وبسلاحهم"-"and by their weapon" has four POSs. This make the task of POS tagging is more complicated than other languages.

In this work, we try to use new method for POS tagging which has prosperities of HMM and N-grams to used for long Arabic sentence with high precision.

2. Related Work

POS tagging is wide studied for many natural languages therefore we will focus on the related work for Arabic Language.

K. Darwish, A. Abdelali and H. Mubarak[4]introduced the use of stem templates as a feature to improve POS tagging by 0.5% and to help ascertain the gender and number of nouns and adjectives. They used PATB 1,2 and 3 corpus and its tag set after simplification.

Mona Diab, Kadri Hacioglu and Daniel Jurafsky[5] presented a Support Vector Machine(SVM) based approach to automatically tokenize (segmenting off clitics), part-of speech (POS) tagging and annotate base phrases (BPs) in Arabic text. They adapted tools that have been developed for English text and applying them to Arabic text. The SVM-POS tagger achieves an accuracy of 95.49%. They used Arabic Penn Tree Bank corpus of 4519 sentences.

Mohamed Attia , Mohsen A. A. Rashwan[6]described a " large-scale – Arabic POS tagger" with private tags set. They show how the resulting POS tags sequences are statistically utilized to train and infer the syntactic diacritics as part of the process of automatic Arabic diacritization. They used private news-domain corpus of 365,100 words, which are lexically analyzed in ArabMorpho format with manual supervision.

Mohammed Albared and Nazlia Omar [7] presented the preliminary achievement of Bigram Hidden Markov Model (HMM) to tackle the POS tagging problem of Arabic language. In addition, they have used different smoothing algorithms with HMM model to overcome the data sparseness problem. Several lexical models have been defined and implemented to handle unknown word POS guessing based on word substring. The average overall accuracy for this tagger is 95.8% using private corpus of 26631 manually annotated.

Saib Mansour - Khalil Sima'an-Yoad Winter [8]proposed an enhanced Part-of-Speech (POS) tagger of Semitic languages that

treats Modern Standard Arabic (henceforth Arabic) and Modern Hebrew (henceforth Hebrew) using the same probabilistic model and architectural setting. They started out by porting an existing Hidden Markov Model POS tagger for Hebrew to Arabic by exchanging a morphological analyzer for Hebrew with Buck walter's (2002) morphological analyzer for Arabic. The accuracy was 96.12% where the used dataset was Arabic Treebank 1,2 and 3.

Fatma Al Shamsi, Ahmed Guessoum[9]they saw that the POS tagger resolves Arabic text POS tagging ambiguity through the use of a statistical language model developed from Arabic corpus as a Hidden Markov Model (HMM). They presented the characteristics of the Arabic language and the POS tag set that has been selected. They then introduced the methodology followed to develop the HMM for Arabic. POS tagging. They used HMM POS tagger achieved a performance of 97% using Arabic Tree Bank corpus of 734 news articles.

Mohammed Albared and Nazlia Omar[10]they presented a study aiming to find out the appropriate strategy to develop a fast and accurate Arabic statistical POS tagger for a limited amount of training material is available. Different configurations of a HMM tagger are studied as well as different smoothing techniques on two small training corpora. The first corpus includes about 29300 words from both Modern Standard Arabic and Classical Arabic. The second corpus is the Quran Arabic.

Aliwy[11] proposed new method called Master-Slaves Technique, which can combine Hidden Markov Model (HMM) tagger as master and maximum match (MM) and Brill taggers as slaves. The main property of this method is that the master tagger will process search sentence with different probabilities (different knowledge). He used private Arabic data set consists of 45 files (29k words) annotated by hand with very rich tagset. The accuracy was 90.05%.

Aliwy[3] used hybrid method by combining rules-based and statistical methods. Three taggers, Hidden Markov Model (HMM), maximum match and Brill taggers are combined by a new method with hand crafted rule based. He used private Arabic data set consists of 45 files (29k words) annotated by hand with private tagset. The accuracy was 92.86%.

3. Theory Background

Ngrams and HMM approaches will be discussed, in this section, because they are used in our suggested approach for POS tagging.

N-Gram

N-gram is conclusive in many NLP tasks such as POS tagging. It is, in general definition, a neighboring sequence of n items. N-grams tagging method uses the probability of previous context (tags) for the current word to be known its POS tag. The value of n can be 1 (unigram), 2(bigram) 3(trigram) or any other value. Unigram is very simple, does not take any tag sequence information into account. It means the word is independent and the tag which has the high probability will be selected for the word. Bigram uses more information by taking the previous tag into account, i.e. the current word tag depends on the tag of previous word only. Trigram adds even more by taking two previous tags into account, i.e. the current word tag depend on the tags of the previous two words. The limitation of this technique is by selecting tag of one word per a time according to the previous word tags only. The following equations are used for unigram, bigram, and trigram POS tagging.

Unigram simplification (tag of the current word i)

$$\hat{t}_i = \arg\max_{t_j} \quad p(w_i \mid t_i) \quad \text{.....(1)}$$

Bigram simplification (tag of the current word i)

$$\hat{t}_i = \arg\max_{t_j} \quad p(w_i \mid t_i)\, p(t_i \mid t_{i-1}) \quad \text{...(2)}$$

Trigram simplification (tag of the current word i)

$$\hat{t}_i = \arg\max_{t_j} \quad p(w_i \mid t_i)\, p(t_i \mid t_{i-2} t_{i-1}) \quad \text{...(3)}$$

Where $p(w_i \mid t_i)$ is the probability of tag t_i given word w_i, $p(t_i \mid t_{i-2} t_{i-1})$ is probability of occurring tag t_i after the previous two tags $t_{i-2} t_{i-1}$. The interpolation parameters can be used for smoothing the rare data and many probabilities in trigram is zero's:

$$p(t_i \mid t_{i-2} t_{i-1}) = \lambda_1 p(t_i) + \lambda_2 p(t_i \mid t_{i-1}) + \lambda_3 p(t_i \mid t_{i-2} t_{i-1}) \quad (4)$$

Where $\lambda_1 + \lambda_2 + \lambda_3 = 1$.

HMM

HMM is used for tagging one complete sentence at a time by choosing the most likely series of tags for thegiven sequence of words. In HMM, the POS problem can be defined as find the best tag sequence t^n given the word sequencew^n.The label sequence t^n generated by the model is the one which has highest probability between all the probable label sequences for the input word sequence. HMM taggers make two additional facilitation assumptions. The first is that the eventuality of a word appearing depends only on its own tag and is independent of neighboring words and tags. The second assumption, is the bigram assumption, is that the eventuality of a tag is dependent only on the previous tag. the following equation is used in HMM POS tagger.

$$t_1^n = \arg\max_{t_1^n} \quad p(t_1^n \mid w_1^n) \approx \arg\max_{t_1^n} \prod_{i=1}^{n} p(w_i \mid t_i)\, p(t_i \mid t_{i-1})$$

4. The Suggested Approach

The sentence in Arabic language is very long in length compared by English language. Itis not known, in some situation, in their limits. Therefore, we suggest new methodology for dealing with Arabic language sentence. This method is considered hybrid between n-gram and Hidden Markov Model (HMM).It takes the characteristics of both cases (n-gram and HMM) which is considered a point of strength.

The new method works by partitioning the sentence into constant parts (windows) in length *n*. theconsecutivewindows will be overlapped by *v* of words where $1 \le v \le 3$, see figure 1. All words in each window will be tagged using HMM tagger without taking in account the beginning and ending of the sentence in order to not affect the calculations. After completing tagging of the whole sentence then, the overlap words between windows w_i and w_{i+1} will be tagged using unigram, bigram or trigram according to *v* value. In many cases the bigram and trigram may have zero's, the equation 4 will be used for smoothing.

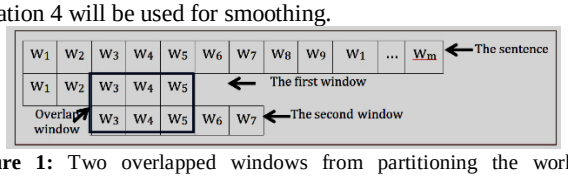

Figure 1: Two overlapped windows from partitioning the working sentence

The method is very simple and can be used for Arabic language for avoiding the problem of calculating probability of long path (tags sequence for very long sentence) in HMM which cause Cumulative errors. These cumulative errors are appears because HMM select the path which has the highest probabilities result from multiplying of the component of the path and hence if the previous component is selected wrongly then it will affect the other components (tag sequence). The suggested approach can be used with any POS tagging method used to tagging whole sentence not only with HMM.

Other benefit from this approach when some text in Arabic corpora did not have the sentence ending and beginning which cause problem in using HMM tagger; therefore our approach can be used safely because it effects on a particular section in a particular sentence and does not affect the rest of the sections in the same sentence.The suggested approach can be implemented as in algorithm1. Data set

Algorithm 1: Improved POS tagging
Input: *S: sentence to be tagged.*
Output: *T: the best sequence of tags.*
Step 1*: TA= array of size (length of sentence) of record (tag1, tag2): to make each overlapped word to have two tags.*
Step 2*: HMM is learned Hidden Markov model tagger;*
NG is learned (Unigram model, Bigram model and Trigram model)
Step 3*: let w = overlapped parts of the sentence of length n where The overlap size = v*
Step 4*: for each part p in w:*
Tag all words in part p using HMM and put the result in TA with keeping the previous estimated tags for the overlapped words
Step 5*: for each overlapped part p in w:*
Select the best tags for each overlapped word using NG
Step 6*: T= TA[1..n].tag1: values of first tag for each word from TA*
Step 7*: return T*
End

Our approach was tested on part of KALIMAT corpus which is multipurpose corpus.it has 29291 articles from al Alwatan-2004 which collected by Abbas [12]. The authors of this corpus did not show that the corpus is golden standard or not for annotation of POS tags. Frm analyzing samples of the corpus, we see that it has many errors this means it is not a golden standard corpus for POS tagging system but it can be used as baselines for other POS taggers comparing with Stanford POSTagger[13] which is used to annotate this corpora. We corrected one thousand articles from this corpus to be as golden standard for our test. It has 21845 sentences of 526321 tokens including punctuations.

5. Results

All the three POS taggers were implemented, in this work, using 10-fold-cross validation. Each fold consists of 900 file as training and 100 files as test for each method. The results of the three classifiers explained in Table 1. Table 2 shows the result of implementation of the suggested method with different window sizes (*w*) and different overlap sizes (*v*). The selected sizes of windows are 6, 10 and 15 respectively with overlap window sizes 1,2 and 3. The results show that our approach is better than traditional HMM. [14][15]

Table 1: Results of Implementing Trigram, HMM and the Suggested method

Fold	Training articles	Testing articles	#test tokens	F-measure %		
				Trigram	HMM	suggested
1	900	100	50963	0.882	0.928	0.965
2	900	100	53181	0.861	0.878	0.933
3	900	100	47243	0.894	0.943	0.977
4	900	100	67875	0.874	0.915	0.962
5	900	100	52106	0.907	0.934	0.944
6	900	100	47838	0.884	0.925	0.957
7	900	100	57192	0.906	0.932	0.963
8	900	100	41931	0.883	0.926	0.973
9	900	100	49561	0.898	0.943	0.946
10	900	100	58431	0.892	0.935	0.955
Average				0.888	0.925	0.957

Table 2: Results of the Suggested Method with Different Window Sizes (w=6,10,qnd 15) and Different Overlap Sizes (v=1, 2 and 3)

Fold	#test tokens	F-measure								
		w=6			*w*= 10			*w*= 15		
		v=1	*v*=2	*v*=3	*v*=1	*v*=2	*v*=3	*v*=1	*v*=2	*v*=3
1	50963	0.913	0.93	0.942	0.942	0.956	0.965	0.928	0.939	0.954
2	53181	0.885	0.897	0.907	0.910	0.925	0.933	0.896	0.909	0.915
3	47243	0.865	0.927	0.952	0.899	0.949	0.977	0.880	0.930	0.963
4	67875	0.901	0.934	0.936	0.932	0.960	0.962	0.914	0.946	0.949
5	52106	0.893	0.910	0.918	0.920	0.937	0.944	0.904	0.917	0.931
6	47838	0.897	0.919	0.929	0.928	0.949	0.957	0.913	0.939	0.938
7	57192	0.914	0.933	0.940	0.943	0.961	0.963	0.925	0.949	0.951
8	41931	0.926	0.936	0.946	0.952	0.964	0.973	0.936	0.952	0.960
9	49561	0.890	0.917	0.917	0.925	0.940	0.946	0.913	0.924	0.926
10	58431	0.902	0.931	0.932	0.934	0.951	0.955	0.915	0.934	0.940
Average		0.899	0.923	0.932	0.929	0.949	0.958	0.912	0.934	0.943

6. Conclusion

We made a new suggested method for implementing HMM tagger to be compatible with long sentences in Arabic language. The suggested approach was implemented on 1000 documents edited manually. The implementation was doing using HMM, n-grams, and the suggested approach on the same data. The suggested approach was tested using different windows size (w=6, 10 and15) and different overlap size (v=1, 2, and 3). From the result shown in table 1 & table 2, the suggested approach give a higher accuracy than HMM and n-grams. This ensures that using tagging for one complete long sentence at a time will produce cumulative errors in tagging and hence reducing the accuracy.

We recommended that using the suggested approach on different data set and different language for ensuring its compatibility in multiple environments.

References

[1] Jurafsky D & Martin J, "Speech and Language Processing: An introduction to natural language processing", *computational linguistics, and speech recognition*, (2008).

[2] Nitin I & Fred J, *Handbook of Natural Language Processing, Second Edition*, Chapman & Hall/CRC Machine Learning & Pattern Recognition, USA, (2010).

[3] Aliwy AH, "Arabic morphosyntactic raw text part of speech tagging system", *Ph.D dissertation, University of Warsaw, warsaw, Poland*, (2010).

[4] Darwish K, Abdelali A & Mubarak H, "Using Stem-Templates to Improve Arabic POS and Gender/Number Tagging", *LREC*, (2014), pp.2926-2931.

[5] Diab M, Hacioglu K & Jurafsky D, "Automatic tagging of Arabic text: From raw text to base phrase chunks", *Proceedings of HLT-NAACL:Short papers*, (2004), pp.149-152.

[6] Attia M & Rashwan M, "A large-scale Arabic POS tagger based on a compact Arabic POS tags set, and application on the statistical inference of syntactic diacritics of Arabic text words", *Proceedings of the Arabic Language Technologies and Resources Int'l Conference*, (2004).

[7] Albared M, Omar N, Ab Aziz MJ & Nazri MZA, "Automatic part of speech tagging for Arabic: an experiment using Bigram hidden Markov model", *International Conference on Rough Sets and Knowledge Technology*, (2010), 361-370.

[8] Mansour S, Sima'an K & Winter Y, "Smoothing a lexicon-based POS tagger for Arabic and Hebrew", P*roceedings of the Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources*, (2007), pp.97-103.

[9] Surendar, A., & Nelakuditi, U. R. (2017). Editorial -New developments in electronics, cloud and IoT. Electronic Government, 13(4).

[10] Albared M, Omar N & Ab Aziz MJ, "Developing a competitive HMM arabic POS tagger using small training corpora", Asian Conference on Intelligent Information and Database *Systems*, (2011), pp.288-296.

[11] Aliwy AH, "Combining POS taggers in master-slaves technique for highly inflected languages as Arabic", *International Conference on Cognitive Computing and Information Processing*, (2015), pp. 1-5.

[12] Abbas M, Smaili K & Berkani D, "Evaluation of Topic Identification Methods on Arabic Corpora", *Journal of Digital Informa0on Management*, Vol.9, No.5, (2011), pp.185-192.

[13] Toutanova K, Klein D, Manning CD & Singer Y, "Feature-Rich Part-Of-Speech Tagging With a Cyclic Dependency Network", *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, (2003), pp.173–180.*

[14] Z Iskakova, M Sarsembayev, Z Kakenova (2018). Can Central Asia be integrated as asean? Opción, Año 33. 152-169.

[15] G Cely Galindo (2017) Del Prometeo griego al de la era-biós de la tecnociencia. Reflexiones bioéticas Opción, Año 33, No. 82 (2017):114-133