

# Hidden Markov-based Part-of-Speech Tagger for Igbo Language

**Iheanetu, Olamma**  
Department of Computer  
and Information Science  
Covenant University  
Ota, Nigeria

olamma.iheanetu@covenant  
university.edu.ng

**Michael, Kingsley**  
Ecumenical Technology  
m.kingsley90@outlook.co  
m

**Ojo, Sunday O.**  
Inclusive African  
Indigenous Language  
Technology Institute  
Pretoria, South Africa

prof.Sunday.ojo@afriilt.  
institute

## Abstract

Igbo is a resource-scarce Nigerian African language of Bantu language phylum, lacking electronic linguistic resources in sufficient quantity and quality for the development of human language technologies. Developing Natural Language Processing (NLP) pipeline tools for such a language could be challenging, due to the need to balance the linguistics semantics robustness of the tool with computational parsimony. A Part-of-Speech (POS) tagger is a challenging NLP tool to develop for the language because of its morphological richness poses computational linguistics challenge that could affect the effectiveness of the entire NLP system. In this paper, the experience in developing a POS tagger for the language using the Hidden Markov Model (HMM) is presented. It is an on-going project, developed using a small corpus. The results give an approximate accuracy score of 73%, which needs to be improved upon.

## 1 Introduction

A Part-Of-Speech (POS) tagger is a NLP pipeline tool that inputs text from a source language and assigns a part of speech tag to each word in the text, classifying each as noun, verb, adjective, and so on, or a refinement. POS tagging, sometimes referred to Word Category Disambiguation (WCD) involves giving a word in a text, a unique tag based on the word context and grammatical function. Adjacent and related word to the word of interest plays a huge role in disambiguating the word category, enabling automatic text processing in a language.

POS tags can also be employed for grammatical or lexical pattern searches. In any POS tagging assignment, the aim is to identify the morphosyntactic class of each occurring word based on lexical and contextual information. Hence, it is possible that in different contexts, a

given word may identify with two or more morphosyntactic classes. This scenario informs the importance of engaging human linguistic experts of languages in the inundating task of manually tagging study corpora; an unpopular venture, given the large amounts of data and time needed for such tasks.

For languages where homonyms, especially homographs are prevalent, it becomes pertinent to employ a POS tagger to distinguish between such word occurrences. For example, the word *face* in English could be either a noun or a verb depending on its usage in a sentence. Homonyms also occur in Igbo and are most prevalent especially when diacritics are missing in the text. For example, consider the following homographs in Igbo - *akwa* (cry) [H-H], *àkwà* (bed) [L-L], *àkwa* (egg) [L-H], and *akwà* (cloth) [H-L] all have different meaning in Igbo. The diacritics introduce a measure of distinction and without the diacritics, deciphering the meaning of such words would rely heavily on the context in which they are used, or a POS tagger in the language can be used to disambiguate the words. For homophones, ambiguities must be resolved in order to understand the intended meaning of such words. Otherwise, the ambiguities become misleading especially if a Text-to-Speech (TTS) synthesizer is involved. In reality, most Igbo texts are published without the necessary diacritics due to the unavailability of input tools for such symbols.

Due to the increase in amount of computer readable texts, POS taggers have become very useful and indispensable in computational studies of natural languages. Without automatic text processing like POS tagging, it would require thousands of human hours to manually tag texts, especially when a relatively large corpus is involved. Furthermore, manually tagged corpus is not scalable.

Basically, a POS tagger *learns* from a training set of data that has been manually annotated so that it can automatically tag unseen words appropriately. The tagger also learns the context

in which words are used in order to assign appropriate task. In learning word context, adjacent and related words play a crucial role. POS taggers are as accurate as the training data from which they *learn*.

Part of speech taggers for each language can be mutually unrelated tools and each one can use different tools, techniques, and computational models, as may be dictated by the nuances of the lexical semantic system of each language. Apart from those, there are also multilingual tools which can be trained to process more than one language. The core software stays the same, but a different annotation is used for each language.

## 2 Related works

### 2.1 Background to Igbo Language

The Igbo language is one of the Nigerian languages, spoken by the Igbo in South-east Nigeria. The population of Igbo speakers has been put at varying figures by different studies. National Population Commission (2006) estimates an approximate fifteen million Igbo people from the 2006 census; Igbo Open Source project quoted twenty-five million Igbos (<http://igbo.sourceforge.net>), while Central Intelligence Agency (CIA), U.S.A. (2008) reported a population between twenty-four and twenty-five million Igbos. One to two million other Nigerians speak Igbo as a second language in addition to another three to five million people in Diaspora (Linux, 2010).

Approximately thirty dialects of Igbo exists (UCLA, 2009), some of which are spoken in Abia, Anambra, Enugu, Ebonyi and Imo States, all in the eastern part of Nigeria. Some of these dialects include: Umuhija, Onitcha, Orlu, Ngwa, Afikpo, Nsa, Oguta, Aniocha, Eche, Egbema, Oka (Awka), Bonny-Opobo, Mbaise, Nsuka, Ohuhu and Unwana dialects (UCLA, 2009).

Igbo language is a member of the West Benue-Congo languages. Blench and Dendo (2003) formerly classified under the Niger-Congo Kwa language family; a language family that is characterized by high and low tones in which different meanings are applied to the same set of phones (Gale Group, 1999). The language exhibits a rich agglutinative morphology (UCLA, 2009 and Osuagwu, Nwaozuzu, Dike, Nwaogu, and Okoro, 1997). Igbo features a wide variety of

highly productive concatenative and non-concatenative morphological processes. Cascaded affixation is a common occurrence in Igbo morphology owing to the agglutinative nature of the language and it is also highly productive in the language.

### 2.2 Igbo Computational Studies

Due to the increase in the digital textual document and the subtle pressure from the information society to develop human language technologies for computational language studies, Natural Language Processing (NLP) has become indispensable for automatic language processing. The basic resources needed for automatic language processing is computer readable text in a source language. Most resource scarce languages lack this basic requirement and as a result, computational studies of such languages are either slowed down or impeded. Igbo language is among the worlds' less studied languages or resource scarce language because vast electronic linguistic data in the language do not exist.

However, modest efforts have been made in recent times to subject Igbo to computational analysis. Such efforts, as Igbo morphological analyzers (Ayogu, Ignatius, Adetunmbi, Adebayo, Kamelu and Nkiru, 2013), (Iheanetu, and Adeyeye, 2013) and (Iheanetu, 2015), POS tagger (Onyenwe, Onyedinma, Aniegwu and Ezeani, 2019), have recorded some level of successes. Notwithstanding, the language still begs for more efforts towards computational studies in the language.

Iheanetu, (2015) developed an Igbo morphological analyzer using a relatively small corpus and a frequent pattern-based technique. The resulting segmented words had *word label* segments instead of the conventional syntactic tags. Onyenwe, Hepple, Chinedu and Ezeani (2018) and Onyenwe, Onyedinma, Aniegwu and Ezeani (2019) developed a POS tagger for the language using a modified version of the EAGLE tagset to realize 59 distinct tags. However, they propose the employment of an automatic morphological segmentation in order to realise a more fine-grained tagset for Igbo.

Other recent studies like (Ayogu, Adetunmbi and Ojokoh, 2018) tried to deploy a machine translator for English- Igbo, English-Yoruba and Igbo-Yoruba. On the average, the study was able

to achieve meaningful translation between the languages as depicted by the individual BLEU scores. However, the machine translator may need to be improved in order to achieve higher BLEU scores with a lower limit of 50.0.

Some generic Open source POS taggers already exist, which boast of their scalability to any language by just re-training the tagger in the source language. These include, the Stanford Log-Linear POS tagger is one of such tools. It was originally developed for French, English, German, Chinese and Arabic languages (<https://nlp.stanford.edu/software/tagger.shtml>).

They also include Python NLTK and Apache OpenNLP. Both of these are machine learning based toolkit for the processing of natural language texts. They are most commonly used for NLP tasks, such as tokenization, sentence segmentation, part-of-speech tagging, named entity recognition, chunking, parsing, and coreference resolution. However, their effectiveness in use for any language, is contingent on having robust corpus annotations, that accurately capture the nuances of the lexical semantic system of the language. Here lies the challenge with under-resourced language such as the Igbo language, the paucity of such annotated corpora.

### 2.3 Ambiguity in Igbo POS Tagging

The required parallel corpus for POS tagging is not available for Igbo language, hence, the decision to use a translation of the English Bible, which was translated to Igbo using Google API. The resulting text was not consistent with the Onwu orthography which is the official orthography for Igbo. In addition, the morphology did not fully align with the official Igbo morphology.

Igbo language has a rich agglutinative morphology (UCLA, 2009), which sometimes is expressed in cascades of affixation involving mostly extensional suffixes/ enclitics. Cascaded affixation is a very productive morphological process in Igbo (Iheanetu, 2015). This informs the need to employ morphological segmentation in achieving a tagset for the language given that some compound words could be read as short phrases. For example consider the Igbo word *abanyekwalarii* meaning “has entered a long time

ago”. A morphological segmentation of the word will reveal the morphemes that make up the word.

- *abanyekwalarii* → *a* (Prefix) - *banye* (Verb) – *kwa* (Extensional suffix) – *la* (Enclitic) – *rii* (Enclitic).

Prefixation, suffixation, interfixation, compounding and (root word) modification are the common broad morphological processes in Igbo. As simple as these processes may appear, some of them show a level of complexity, owing to some peculiarities of Igbo language like the concept of *vowel harmony*.

In addition, the high level of agglutination in the language presents some peculiar challenges for POS tagging. The English phrase, ‘must eat completely’ (three words in English) is agglutinatively written *richariri*. where *-ri* is verb root (eat), and *-cha* and *-ri.-ri*. are suffixes indicating completion and compulsion respectively.

In the absence of the necessary diacritics, it becomes difficult to differentiate between homonyms. Unfortunately, most Igbo texts are written without these necessary diacritics which are high [ˈ], low [ˌ] tones and downsteps [ˑ] accents for the vowels and syllabic nasals. However in written texts, only the low and midtones are marked (Green and Igwe 1963) in order to facilitate smooth reading and also to make the text wieldy.

## 3 POS Tagset Design

The Penn Treebank tagset was used for the purposes of this study. In total, the tagset had thirty-six tagsets. However, it was observed that some of the tags did not occur in Igbo (for example, article does not exist in Igbo) while most occurring tags in Igbo were missing. For example, *o* could be used for a personal pronoun *he/ she* or could mean *it*, when it is functioning as an impersonal pronoun. Igbo Particularisers (*nke a* and *nke ahu*) were not captured in the tags. Therefore, the original Penn Treebank tagset was modified with the addition of the tag IP to capture impersonal pronouns, with the intention of incorporating many others in the future.

### 3.1 POS Tagging Method

This section discusses the methods and tools used for design and implementation of the Igbo tagset. However for the alignment, no software was used due to corpus paucity.

### 3.2 Data Source

This study is on-going, and the results presented here are part of the preliminary results of investigations carried out. Large portions of the text used for this study were sentences from an English Bible [<http://bibledatabase.com>]. Each verse of the bible in Genesis chapter 1 were tokenized and the tokens were then translated to Igbo using the Google API. Afterwards, the generated Igbo tokens were then manually annotated. In addition to texts from the Igbo Bible, sentences were obtained from twenty newsgroups (<http://people.csail.mit.edu/jrennie/20Newsgroups>), to accommodate patterns that lace everyday language use. The dataset realized from the outlined sources was relatively small, producing a total of nineteen (19) sentence tokens. Out of this number, twenty percent (20%) was used as test set (4 sentence tokens) while eighty (80%) was used to train the tagger (15 sentence tokens). The Penn Treebank (Marcus, Santori and Marcinkiewicz, 1993). POS tagset was used for the classification and this tagset includes numbers and punctuations tags. However, we observed that the Penn Treebank tagset did not capture all morphosyntactic classes in Igbo, hence we introduced a morphosyntactic class in the tagset used for the classification. Many more will be likely introduced before the completion of this work. See Table 1.

### 3.3 The HMM-based Method

The probability-based Hidden Markov Model (HMM); was used for predictive pattern modeling of Igbo POS. HMM is structured to look for the probability of a sequence given an observation:

$$P(S|O) = \frac{P(S,O)}{P(O)}$$

The sequences; (S) represents the tags while the observations (O) represents the sentence tokens.

TABLE I. MODIFIED PENN TREEBANK TAGSET FOR IGBO

No.	Tag	Description
1	CC	Coordinating conjunction
2	CD	Coordinating conjunction
3	DT	Determiner
4	EX	Existential there
5	FW	Foreign word
6	IN	Preposition or subordinating conjunction
7	JJ	Adjective
8	JJR	Adjective, comparative
9	JJS	Adjective, superlative
10	LS	List item marker
11	MD	Modal
12	NN	Noun, singular or mass
13	NNS	Noun, plural
14	NNP	Proper noun, singular
15	NNPS	Proper noun, plural
16	PDT	Predeterminer
17	POS	Possessive ending
18	PRP	Personal pronoun
19	PRPS	Possessive pronoun
20	RB	Adverb
21	RBR	Adverb, comparative
22	RBS	Adverb, superlative
23	RP	Particle
24	SYM	Symbol
25	TO	to
26	UH	Interjection
27	VB	Verb, base form
28	VBD	Verb, past tense
29	VBG	Verb, gerund or present participle
30	VBN	Verb, past participle
31	VBP	Verb, non-3rd person singular present
32	VBZ	Verb, 3rd person singular present
33	WDT	Wh-determiner
34	WP	Wh-pronoun
35	WP\$	Possessive wh-pronoun
36	WRB	Wh-adverb
37	IP	Impersonal pronoun

Therefore, the model looks for the best sequences combinations that maximizes P(S|O): To maximize the probability sequence:

$$P(s_1 \dots s_n | O_1 \dots O_n) \\ = P(s_1 | s_0) P(o_1 | s_1) P(s_2 | s_1) P(o_2 | s_2) \dots$$

For N observations and K states, there are  $K^N$  sequences, and the larger N is the more recursive steps needed in the calculations. Therefore, the use of dynamic programming (shortest path/ tree search algorithms) to arrive at a solution was employed. A Dynamic programming algorithm commonly used with HMM is Viterbi, which attempts to solve the recursive problem:

$$v_i(s_{i=x}) = \max_{k=1}^L [v_{i-1}(k) \cdot P(x|k) \cdot P(o_i|x)]$$

The variable  $v_i(x)$  represents the maximum probability that the  $i$ -th state is  $x$ , given that  $O^i_1$  has been seen. At each step, a record of back pointers showing which previous state led to the maximum probability was taken

$$S_{best} = \arg \max_s \frac{P(S, O)}{P(O)}$$

## 4 Evaluation

The training and test set sentence tokens was randomly picked by shuffling the dataset using python script. Accuracy measure was calculated thus:

Accuracy = number of correct tags / number of words

Test accuracy with 13 sentence tokens gave an accuracy of 66.67% .

The error rate was calculated with the formula:

$$\text{error rate} = 1 - \text{Acc}$$

Therefore:

$$\text{Error rate} = 1 - 73.33\%$$

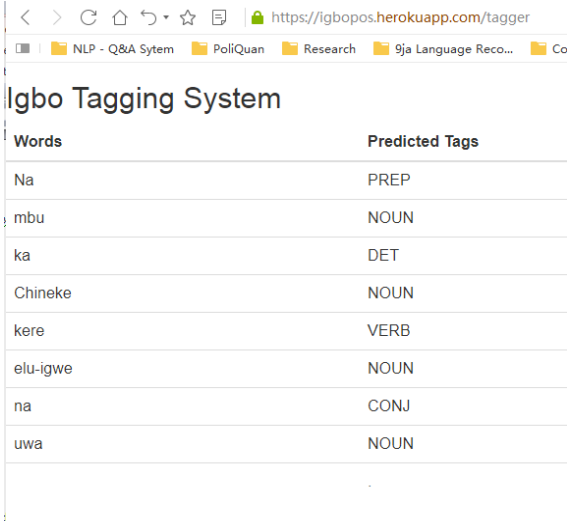
$$\text{Error\_rate} = 1 - 0.7333$$

$$\text{Error\_rate} = 0.2667$$

A demo prototype was put up online (<https://igbopos.herokuapp.com/>) for further test of the algorithm on new sentence tokens and for dataset gathering for constant upgrade of the performance of the model.

The accuracy of the alignment process had a great impact on the overall accuracy of the tagger. It was observed that some words in the source language (English) were captured by two or more

words in the target language (Igbo) and vice versa. Also, the inconsistency with the official Igbo orthography was a major downside of the resulting translation. Some of the words used for translation were either not necessary or was inappropriate. However, the major challenge faced in this study was to manually annotate/ tag the translated Bible verses in order to realize a sufficiently large amount of tags (parallel corpus) to train the Igbo tagger with. Given the short time available for this exercise, it was not possible to realize the desired number of tokens, hence only 19 tokens were used for the alignment and subsequently, to train the tagger. With more tokens and fine-grained tags, it is very possible that the accuracy of the tagger would greatly increase.



Words	Predicted Tags
Na	PREP
mbu	NOUN
ka	DET
Chineke	NOUN
kere	VERB
elu-igwe	NOUN
na	CONJ
uwa	NOUN

Figure. 1 Screenshot of POS Tagger output

## 5 Conclusions, Limitations and Futrure works

The study tried to develop an Igbo POS tagger using 19 tokens generated from a corpus consisting the first chapter of the Igbo Bible and a translation of the same using Google API. The resulting translation was not consistent with the official Igbo orthography which is the Onwu orthography and also, sometimes, the accepted morphology of Igbo. The Penn Treebank tagset used did not capture all word forms in Igbo and as such, may need to be modified in order to accommodate the morphological peculiarities of



Igbo. For this study, only one new tag was introduced, among the many that were missing. Hence, more efforts need to be geared towards achieving a suitable tagset for training an Igbo POS. A possible direction may be to employ morphological segmentation as suggested by Onyenwe *et al.*, 2018.

This is an on-going project and the preliminary test results presented here demonstrate success in the chosen tools for investigation. The researchers hope that adequate amount of data will be generated when the tagger is constantly tested online. In addition, the criticisms will provide a positive feedback for the improvement of the tagger.

## References

- Ayogu, I., Ignatius, I., Adetunmbi, Adebayo, O., Kamelu and Nkiru, C. 2013. Finite state concatenative morphotactics: the treatment of Igbo verbs. *International Journal of Computing and ICT Research* 7.1:1818-1139.
- Ayogu, I., Adetunmbi, A and Ojokoh, B. 2018. Developing Statistical Machine Translation System for English and Nigerian Languages. *Asian Journal of Research in Computer Science*. 1(4): 1-8, 2018; Article no. AJRCOS.44217
- Blench, R. and Dendo, M. 2003. Language death in West Africa. Retrieved, June 13, 2015, from [www.ling.pdx.edu/childs/DKB.../blench\\_langauge\\_d\\_ath\\_west\\_africa.pdf](http://www.ling.pdx.edu/childs/DKB.../blench_langauge_d_ath_west_africa.pdf)
- Central Intelligence Agency (CIA), U.S.A. 2008. Igbo people. World Fact Book. Retrieved October 20, 2010, from [http://en.wikipedia.org/wiki/CIA\\_World\\_Factbook](http://en.wikipedia.org/wiki/CIA_World_Factbook).
- Gale Group Inc. 1999. Igbo. Junior Worldmark Encyclopedia of World Cultures. Retrieved August 10, 2010, from Encyclopedia.com: <http://www.encyclopedia.com/doc/1G2-3435900354.html>
- Green, M. and Igwe, E. 1963. A descriptive grammar of Igbo. Berlin: Akademie Verlag.
- Igbo Open Source Translation Project. Retrieved October 19, 2010, from <http://igbo.sourceforge.net/>
- Iheanetu, O. and Adeyeye, M. 2013. Finite state representations of reduplication processes in Igbo. *IEEE Xplore Digital Library*. doi:10.1109/AFRCON.2013.6757772. 1-6.
- Iheanetu, O. U. 2015. Data-driven model of Igbo morphology. Ph.D thesis. Africa Centre for Information Science (ARCIS), XIV + 208 pages.
- Linux, N. 2010. Igbo open source translation project. Sourceforge.net. Retrieved August 10, 2011, from <http://igbo.sourceforge.net/>
- Marcus, M., Santori, B. And Marcinkiewicz, M. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19:323-330.
- National Population Commission. 2006. Retrieved October 18, 2010 from <http://www.population.gov.ng/index.php/censuses>
- Onyenwe, I. E., Hepple, M., Chinedu, U. and Ezeani, I. M. 2018. A basic language resource kit implementation for the IgboNLP project. *ACM Trans, Asian Low-Resour. Lang. Inf. Process.*, Vol 17, No. 2, Article 10
- Onyenwe, I., Onyedinma, E., Aniegwu, G and Ezeani, I. M. 2019. Bootstrapping method for developing part-of-speech tagged corpus in low resource languages tagset - a focus on an African Igbo. *International Journal on Natural Language Computing (IJNLC)*, Vol.8, No.1, pp. 13-27.
- Osuagwu, B. I. N., Nwaozuzu, G. I., Dike, G. A., Nwaogu, V. N. and Okoro, L. C. 1997. Fundamentals of linguistics. Owerri : Colon Concept Ltd.
- University of California, Los Angeles (UCLA) Language Materials Project. 2009. Igbo. UCLA Language Materials Project. Retrieved October 20, 2010, from <http://www.lmp.ucla.edu/Profile.aspx?LangID=13&menu=004>.