Use of Transformation-Based Learning in Annotation Pipeline of Igbo, an African Language.

Ikechukwu Onyenwe[1], **Mark Hepple**[1], **Chinedu Uchechukwu**[2], **and Ignatius Ezeani**[1]
[1]Department of Computer Science, University of Sheffield, UK
{i.onyenwe, m.r.hepple, ignatius.ezeani}@sheffield.ac.uk
[2]Department of Linguistics, Nnamdi Azikiwe University, Nigeria
neduchi@yahoo.com

Abstract

The accuracy of an annotated corpus can be increased through evaluation and revision of the annotation scheme, and through adjudication of the disagreements found. In this paper, we describe a novel process that has been applied to improve a part-of-speech (POS) tagged corpus for the African language Igbo.

An inter-annotation agreement (IAA) exercise was undertaken to iteratively revise the tagset used in the creation of the initial tagged corpus, with the aim of refining the tagset and maximizing annotator performance. The tagset revisions and other corrections were efficiently propagated to the overall corpus in a semi-automated manner using transformation-based learning (TBL) to identify candidates for correction and to propose possible tag corrections. The affected word-tag pairs in the corpus were inspected to ensure a high quality end-product with an accuracy that would not be achieved through a purely automated process. The results show that the tagging accuracy increases from 88% to 94%. The tagged corpus is potentially re-usable for other dialects of the language.

1 Introduction

When texts and human judgements are stored in computer-readable form, the result is called annotation. Annotation is developed mostly through hand-coded means, so it is important to measure the reliability of the tagset that produced it. The fundamental assumption of this exercise, as discussed in (Artstein and Massimo, 2007; Raquel, 2011), is that the output of manual annotation is considered reliable if it can be computed that annotators are consistent, and the consistency is measured using metrics from the study of Landis and Koch (1977), Krippendorff (1980), and Green (1997). If different annotators produce consistently similar results then we can infer that they have internalized a similar understanding of the tagging scheme, and can expect them to perform consistently under this understanding. The outcome of this exercise is high consistency tagged sub-corpora containing POS-tags described in the tagset.

This paper describes how we leveraged the by-products of the inter-annotation agreement (IAA) exercise to improve the quality of the initial tagged Igbo corpus (ITC0), instead of ignoring them and tagging new text, which saves effort, time and money. A quality tagged corpus can help to maximize the performance of automatic POS-taggers used for tagging similar texts. We employ both manual and automatic processes in a semi-automatic method for this work. Our semi-automatic annotation method uses Transformation-based Learning (TBL) and a human expert, who is involved in several stages of the process. First, an initial Igbo tagged corpus (ITC0) was developed in a distributed manner using the tagset reported in Onyenwe et al., (2014). Through an inter-annotation agreement (IAA) exercise, this tagset (TS0) was evaluated and revised to ensure a more reliable and reproducible result. Then we use TBL to find and propagate changes from the IAA to this initial tagged corpus in an automated manner; an expert human annotator verifies locations TBL has marked for changes instead going through the entire text. Through this semi-automated process, the quality of the tagged corpus is increased with minimum expense. TBL is suitable for this because its inductive method performs very well using annotated corpora whose sizes are smaller than that of n-gram models, and it is an error-driven learner.

Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects, pages 24–33,
Hissar, Bulgaria, September 10, 2015.

TBL is a machine learning (ML) algorithm originally developed by Brill (1992). It starts with an initial state and requires a correctly tagged text, called *truth*, for training. The training process iteratively acquires an ordered list of rules that correct the errors found in the initial state until this initial state resembles the truth to some acceptable degree.

2 The Igbo Language

Igbo is one of the major languages spoken in eastern Nigeria by about 32 million native speakers[1]. It has been classified as a Benue-Congo language of the Kwa sub-group of the Niger-Congo family[2]. It adopts the Ọnwụ Committee orthography[3] and has 28 consonants and 8 vowels. Nine of the consonants are digraphs and the vowels are divided into two harmony groups that are distinguished on the basis of the Advanced Tongue Root (ATR) phenomenon (Uchechukwu, 2008). The majority of the words of the language select their vowels from the same harmony group. There are 3 distinct tones recognized in the language, *High* [´], *Low* [`], *downstep* [¯] (Emenanjo, 1978; Ikekeonwu, 1999). The tonal features of the language could be lexical or grammatical. For example, at the word level, *akwa* could mean 'bed/bridge', 'cry', 'cloth', or 'egg', but can be disambiguated with tones, as follows: akwa "cry", akwà "cloth", àkwà "bed or brigde", àkwa "egg". At the grammatical level, an interrogative sentence is distinguished from a declarative sentence through a change in tone (e.g. *ọ ga-abịa* "He will come", *ọ̀ ga-abịa?* "Will he come?"). Igbo is an agglutinative language in which its lexical categories undergo affixation, especially the verbs, to form a lexical unit. For example, *erichariri* in word form is made up of 4 morphemes: the verbal vowel prefix "e", verb root "ri", extensional suffix "cha", and a second extensional suffix "ṛịṛị". Its occurrence in the sentence "*Obi must eat up that food*" is illustrated below:

Obi	ga-erichariri	nri	ahụ
Obi	aux-eat.completely.must	food	DET

Igbo word order is Subject-Verb-Object (SVO), with a complement to the right of the head in all types of phrases, for example, *"Okeke killed a snake*" is written:

Okeke	gbu-ru	agwọ
Okeke	kill-rV(Past)	snake

3 Related Work

Finding and correcting errors to make more accurate annotated data as found in Loftsson (2009) and Helgadóttir et al., (2012) and our work are relatively similar in the aspect of inspecting marked data positions, but entirely different in methods. Loftsson (2009) and Helgadóttir et al., (2012) applied trained POS-taggers singly and combined, respectively, then the outputs were compared with the gold standard and differences found were marked as error candidates for verification. Whereas our method projects changes made in the IAA into the main tagged corpus, and all positions where these changes occurred are inspected further.

4 Building Input States of TBL

TBL makes use of two input states in its contextual module: the initial state and the truth state. Sections 4.1 and 4.2 describe these two input states.

4.1 Corpus Creation and Annotations

The Igbo language resources used for this study are the New World Translation Bible[4] (NWT) and the initial tagset (TS0) described in Onyenwe et al., (2014). For this study, we collected the new testament portion, which is about 260k tokens and 8k sentences. For rapid POS-tagging, chapters in the Bible corpus were allocated randomly to six groups, producing six corpora portions of approximately 45,000 tokens each (see table 1); each annotator annotates one group separately. The resulting output of this shared task is ITC0.

Key features of the initial tagset used to produce ITC0 comprise two parts, 44 POS-tags for non inflected tokens and 15 for inflected tokens. These 15 POS-tags are represented as α_XS for $\alpha \in \{$infinitive verbs, simple verbs, participles, gerunds, auxiliaries, conjunctions, interrogatives, ...$\}$ and _XS for any affixes, and without _XS are collapsed in the 44 POS-tags. The reason behind

[1] http://en.wikipedia.org/wiki/Igbo_people [July, 2015]

[2] http://www.igboguide.org/HT-igbogrammar.htm [July, 2015]

[3] http://www.columbia.edu/itc/mealac/pritchett/00fwp/igbo/txt_onwu_1961.pdf [July, 2015]

[4] Obtained from jw.org.

Group 1	*Matthew, Philemon, 2 Peter, 1 Timothy, 1 Peter*
Group 2	*Acts, 2 Corinthians*
Group 3	*Mark, Revelation, Galatians, 3 John, 2 John*
Group 4	*John, Philipians, James, Colossians, 1 John, 1 Thessalonians*
Group 5	*Luke, Ephessians, 2 Thessalonians, Titus*
Group 6	*Romans, Hebrew, 1 Corinthians, 2 Timothy, Jude*

Table 1: Bible Book Selections by Group

this division is to capture all tokens with and without affixes in the main corpus since Igbo is an agglutinative language, which is a valuable step towards automated morphological segmentation of Igbo. Also found in this TS0 is multiword cases in the nominal class, which is caused by verb nominalization and its inherent complement. Special tags are used to represent this: tags for the verbal and inherent components.

4.1.1 Cleaning the Corpus "ITC0"

Given the six POS-tagged sub-corpora, we collected the best examples and eliminated errors found in the process. In most cases, this process is indistinguishable from "editing". The types of errors found are ambiguous-tag (1st row of table 10; where annotators could not apply a specific POS-tag to a particular token), no-tag (2nd and 3rd rows of table 10; where tokens are not classified by annotators) and wrong-form (4th row of table 10; where valid POS-tags are wrongly represented). POS-tags found in this error set are 39 in number and 5,062 tokens were affected (1.92% of the main corpus). Proper consultations were made with an Igbo linguist to resolve errors in the unspecified-tag and no-tag sets. In solving the remainder, we built a POS-tag replacement dictionary of the errors in the wrong-form class and pass the ITC0 through it to produce ITC1. The POS-tag replacement dictionary is represented as

tag_replacement = {'INT':'INTJ', 'VSI_OVS': 'VSI_XS', ...}

Another issue that caused no-tag error was improper word form. For example, the token *bụla* is incomplete without *ọ*; in the Bible, both were separated by a lexical space *ọ bụla* 'any'. If annotators had assigned *ọ* with a POS-tag 'PRN' (since it has pronoun form), identifying the right POS-tag for *bụla* became challenging since its meaning is incomplete. This was fixed by removing the lexical space between them. The main corpus size which was originally 264,795, after initial tokenization this was reduced to 263,854. Table 2 shows a few examples of tokens affected and solutions provided.

token id	token	error	resolved	total types
12291	ahụkwa	DEM/DEMXS	DEM_XS	138
4	nke	CJN/*	CJN	
26189	mkpịrịkpị	QTF/XXXX	NNQ	
59639	mpịakọta	NOTAG	NNC	156
1717	wit	XXXX	NNC	
58325	bụla	NOTAG	ọbụla/QTF	941
11790	ee	INT	INTJ	3827
815	chọọ	vSI_OVS	VSI_XS	
1073	fụọ	VSI_OVS	VSI_XS	
3537	nwee	OVS	VSI_XS	
7	banyere	VRV_XS	VrV_XS	

Table 2: Different error forms and corrections

4.2 Tagset Revision and Inter-Annotation Exercise

We used human annotators who are both Igbo linguists and native speakers for adding POS-tags to the Igbo text according to the initial tagset (TS0) guideline. There are factors that motivated the revision of TS0 in order to maximize human annotators agreement. The confusing factors we found among human annotators were related to the status of what to call participles, agentive/instrumental nouns, preposition, etc. For example, annotators had issue classifying some verbs when they change their structures as they precede or follow a pronoun. Mostly they chose to tag them participle (VPP) because the changed structure is prefixed *a/e*, which makes them look like participles. The worst case we found was the handling of the nominal class formed through verb nominalization with their inherent complements. There are agentive and instrumental nouns represented in POS-tag as NNAV NNAC and NNTV NNTC respectively, where V and C are the verbal and inherent noun components of the structure which should always appear as a linked pair. For example, *ọgụ/NNAV egwu/NNAC* "singer" and *ngwu/NNTV ji/NNTC* "digger", but link pairs like *ntachi obi* "steadfastness", *nnwere onwe* "freedom", etc are neither agentive nor instrumental nouns. These and many other issues led to evaluation and revision exercise of TS0. To solve the nominal class case, we rede-

fined agentive and instrumental nouns into multi-word nouns (NNCV NNCC), so that all tokens in this class can easily fit in, which results as shown in table 5. We also introduced $alpha_$BPRN tags to clarify the difference between some verbs functions when prefixed with a vowel *a/e* caused by pronoun location on a sentence or caused by preceding auxiliary verbs. For instance, the word *esi* in *Ọ na- esi nri* "He is cooking " and *esi m Sheffield abịa* "I am coming from Sheffield" functions differently. The first is verb participle (VPP) because of the auxiliary verb *na-*, while the second is a simple verb inflected by a vowel prefix *e* as a result of the position of pronoun *m* in the sentence. Therefore, we introduced *VSI_BPRN* tag to indicate that *e* in *esi* is *m*-bound and *BPRN* tag for *m*-bound. It is assigned *VSI* if sentence pattern changes to *m si Sheffield abịa* "I am coming from Sheffield", while *m* is assign *PRN*.

The main objective we assigned to ourselves while revising the tagged corpus and tagset, was to get high quality tagged corpus and a specific tagset appropriate for Igbo and to maximize agreement among human annotators, in order to ensure high consistency of the tagged corpus. However, agreement among human annotators is not a guarantee for tagset quality, otherwise the trivial and uninformative tagset of one POS-tag size would be optimal. Most meaning-carrier words were assigned POS-tags based on the grammatical role they play in a sentence. Nevertheless, the more informative a tagset is, the less the taggers (human and automatic) accuracy tends to be. Therefore, one has to know where to strike a balance between the tagset informativeness and the tags performance. The tagset revision process affected its size because POS-tags were simplified, removed, and added: the size moved from 59 POS-tags to 62 POS-tags and finally to 69 POS-tags. The effects of some TS0 revisions are seen in the table 4.

4.2.1 Inter-Annotation Agreement

The Inter-annotation agreement process took three iterative phases, and four of the six annotators that produced ITC0 were used (two dropped out and another native speaker was employed instead). In each phase, a subset of main corpus was randomly selected. The tagging scheme used was evaluated and revised at each phase. Since there are 5 human annotators ($l1, l2, l3, l4, l5$, where l = linguist), each phase produced 5 annotations of the selected texts, and from these annotated texts we collate standard outputs through voting; for each token, we consider POS-tags with the highest agreement, while ignoring those with total disagreement. We take the collated outputs as our presumed truths, which serves as "silver standard" (SS) against which individual annotators are compared. The quality of the SS is determined by the annotators' tagging consistency calculated using inter-annotation agreement metrics as discussed in section 4.2.2. The SS and annotated texts ($tl1, tl2, tl3, tl4, tl5$) here will serve as TBL truth states in section 5.

	first IAA	second IAA	third IAA
# of sentences	150	150	150
# of tokens	4977	4963	4851

Table 3: IAA texts statistics

4.2.2 Measures

We adopted *Model and guidelines* → *Annotate* → *Evaluate* → *Revise* (M-A-E-R) methodology of (Pustejovsky and Stubbs, 2012), which is an internal part of MATTER annotation development cycle. We iteratively applied this *M-A-E-R* cycle, until all tags contributing huge disagreements in the annotations are corrected resulting in a higher consistency level among annotators. In each phase, the annotations *-A-* by annotators were done independently using our *M-* (model and guidelines). At the end of each phase, we collect all annotations and apply *-ER* (Evaluate and Revise). The whole process took 3 iterations of revision after cleaning and discussion before the final version. In each iteration, randomly selected texts from main corpus of size about 4.5k tokens was used, making a total of about 14k tokens on the whole (approximately 5% of the main corpus), see table 3. Performance was evaluated using f-measure, simple accuracy method and kappas. Our experiment assumed that each token is fully disambiguated, that is, one tag for one token *tok/t*.

In computing agreement, we use f-measure metric to provide a more detailed picture of inter-annotator agreement between annotators on individual parts-of-speech. The f-measure relates to precision and recall in the usual way. For each phase, we find the micro-average precision and recall, then calculate f-measure. In more detail, for the five annotators, given an annotator, *say* $l1$, we calculate its precision relative to silver standard (SS) developed (see section 4.2.1) with respect to a tag *t* in the set *s* of tags used, which is the num-

ber of tokens both SS and $l1$ agree to be *t* divided by the number of times SS say a token to be *t plus* number of times $l1$ has given *t* to a token different from both agreements. This is same calculation for recall only that division is by number of times $l1$ classify a token to be *t plus* number of times SS has given *t* to a token different from both agreements. See results in table 4.

Tag	Precision		
Tag	1st IAA	2nd IAA	3rd IAA
NNC	95.40	96.16	96.65
PRN	99.03	99.70	98.10
PREP	92.89	97.07	99.00
VPP	88.47	89.17	96.62
VSI	90.01	93.10	93.11
VIF_XS	88.96	68.43	95.49
VPERF	52.86	62.10	78.65
	Recall		
NNC	90.62	90.04	95.11
PRN	98.22	99.52	99.06
PREP	94.39	98.60	99.06
VPP	89.51	93.13	95.24
VSI	89.43	90.02	97.49
VIF_XS	58.46	84.38	85.00
VPERF	52.50	75.00	76.00
	f-measure		
NNC	92.31	92.45	95.36
PRN	98.12	99.11	98.07
PREP	93.09	97.32	98.53
VPP	88.04	90.13	95.33
VSI	88.39	90.90	94.71
VIF_XS	61.13	70.84	87.41
VPERF	45.05	59.36	71.59

Table 4: Some POS tags precision, recall and f-measure of first, second and third phases of annotations.

Also, we compute the overall agreement scores in two ways. Firstly, using the cohn's kappas and secondly, simple accuracy. We calculate

Accuracy = $\frac{tp}{N_n}$

where *tp* is true positive for all annotators and N_n is the total number of tokens of all classes combined together since they are same text.

kappas (k) = $\frac{A_o - A_e}{1 - A_e}$

where A_o is observed agreement, A_e is expected change agreement, A_o–A_e is how much agreement beyond chance was found and 1–A_e is how much agreement beyond chance is attainable (Raquel, 2011). So *k* is the proportion of the possible agreement beyond chance that was actually achieved. See results in table 6.

Tag	Precision		
	1st	2nd	3rd
NNAV	51.33	0.0	
NNAC	-	-	
NNTV	0.0	-	
NNTC	0.0	-	
	Recall		
NNAV	80.00	0.0	
NNAC	-	-	
NNTV	0.0	-	
NNTC	0.0	-	
	f-measure		
NNAV	55.52	0.0	
NNAC	-	-	
NNTV	0.0	-	
NNTC	0.0	-	
Solution			
Tag	Precision		
NNCV			77.81
NNCC			81.14
	Recall		
NNCV			73.33
NNCC			73.33
	f-measure		
NNCV			74.27
NNCC			75.79

Table 5: Some WORST POS tags precision, recall, and f-measure and solution proffered.

	Cohn Kappa			Raw agreement		
coders	1st	2nd	3rd	1st	2nd	3rd
l5+l2	81.35	85.49	91.28	83.14	86.78	92.08
l4+l3	91.77	89.23	92.65	92.51	90.17	93.28
l4+l5	83.96	84.57	88.55	85.39	85.92	89.55
l5+l3	83.76	86.08	90.91	85.19	87.27	91.71
l1+l3	84.56	89.60	95.49	86.00	90.53	95.84
l4+l1	86.36	90.09	91.98	87.62	90.99	92.62
l1+l2	84.80	98.71	92.84	86.32	98.83	93.44
l2+l3	85.97	89.27	92.91	87.30	90.23	93.57
l1+l5	84.88	85.17	89.66	86.28	86.50	90.52
l2+l4	86.82	89.44	90.59	88.11	90.41	91.45
Aves	**85.43**	**88.77**	**91.69**	**86.79**	**89.76**	**92.41**

Table 6: IAA scores based on Kappa statistics and simple accuracy formula for the first, second and third annotations.

5 TBL Propagation Method

We have created a satisfactory tagset (and associated guideline) through the revision of TS0. To create a gold standard of the Igbo corpus, which is what to use in training and testing machine learning classifiers, it is expected that those human annotators involved in the tagset revision cycle be used at this level as they have best understanding of the revised tagset to annotate the Igbo corpus afresh or to identify and correct changes on the initial tagged corpus based on the revised tagset, which will consume time and money. Instead we devised automatic method which used by-products of section 4.2.1 (annotated texts $(tl1, tl2, tl3, tl4, tl5)$) and output of section 4.1.1 (ITC1) to propagate changes found in the former to the latter, and flag locations where these changes occurred on ITC1 for inspection. Through this largely automated process, we expect to reduce the amount of human annotator time and effort, by only requiring the attention of a human annotator (the expert) on the marked positions instead of the entire text. Thus the quality of the corpus is increased with a minimum of expense. The approach of requiring that all revisions should be inspected by an expert annotator is needed to ensure a good quality end-product, with an accuracy that could not be achieved through a purely automated process.

There are two stages in this method, firstly, we used the silver standards (SS) developed from the collation of annotated texts $(tl1, tl2, tl3, tl4, tl5)$ (discussed in section 4.2.1) as the TBL truth state and "the corresponding subset" of ITC1 as TBL initial state. We trained a TBL learner on both states and applied these generated rules to the entire ITC1 to find errors on ITC1 and flag affected positions for inspections. The idea here is that the material from ITC1 is in erroneous state, as shown by its differences to the SS. TBL will learn rules to correct these errors. When the same rules are applied elsewhere in the corpus, the location where any rule 'fire' can be seen as candidate instances for of similar errors. All these locations are inspected by a human expert annotator. Since the TBL rule that fires at a location will propose a specific POS-tag change, the human expert can either accept the TBL proposed change, retain the existing tag at the location where the current POS-tag is deemed correct, or impose an alternative change according to his knowledge of revised tagset when neither TBL proposed tag or current tag are correct. For efficient inspections, we used the marked positions to get *word* current tag and contextual information, which helps in facilitating corrections.

1. Get silver standard from IAA to serve as TBL truth state, TS.
2. Take "the corresponding portion" of ITC1 to serve as TBL's initial state, IS.
3. Train TBL model on both TS and IS.
4. Apply TBL generated rules to ITC1.
5. Inspect locations where rules 'fire'.
6. Repeat from step 1 for TS from each phase of IAA.

# of iteration	TBL change accepted	no change	Manual change
1	3663	1215	420
2	1788	376	297
3	11161	3978	2592
	16612	**5569**	**3309**
Total inspected locations: **25,490**			

Table 7: Result statistics after inspection

Table 7 gives detail of inspected flagged positions - the number of TBL changed tags accepted (where the current tag is not correct), rejected, where current tag prevailed, and neither TBL changed tag nor current tag was correct, so we chose from revised tagset. An improvement of 25,490 inspected locations were made on ITC1 with 19,921 effective changes giving ITC^I (improved ITC1).

Among the human annotators used in section 4.2.1, there are some that have better understanding of a particular POS-tag than the others. Therefore, some POS-tags that were voted out in silver standard creation might be correct if found and inspected, In this second stage, we went further to find in each of the annotated texts $(tl1, tl2, tl3, tl4, tl5)$ POS-tags that were not captured in the silver standard used in first stage. That is, finding and inspecting on ITC^I where one annotator's rule triggered and others did not and vice versa. In this experiment, instead of silver standard serving as TBL truth state, we used each of annotated texts $(tl1, tl2, tl3, tl4, tl5)$ and a subset of ITC^I as TBL initial state. The process steps are same with the first stage except line 6: *Repeat from step 1 for TS from annotated texts in*

each IAA phase. In this stage, we find the impact of 1 annotated text by a human annotator ($l1$) weighted against 4 annotated texts of other four ($l2, l3, l4, l5$) on ITC^I. That is, for each TBL trained on both groups ($l1$ and $l2, l3, l4, l5$), we find and inspect word-tag pairs on ITC^I: where one annotator's rule fired and four others did not (grp1), where four annotators' rule fired and one did not (grp2), and where both fired (assigning the same POS-tag; grp3). In summary, out of 41,990 word-tag pairs flagged by this process, 39,151 is where grp1's rule fired, 2,468 where grp2's rule fired and 371 where grp3's rule fired. From these, 12,996 of 39,151, 1,836 of 2,468, and 318 of 371 have been inspected in the previous stage. All locations inspected by human expert are marked never to be inspected again because we believe that human expert judgement supersedes any other one. In whole, 26,839 word-tag pairs were inspected, out of which, effective change of 5,684 for grp1, 76 for grp2 and 6 for grp3 were made on ITC^I to give ITC2.

Note, in the both stages, TBL proposes additional changes, from which new rules can be formed in the next phase. Human annotators used in the tagset revision were not used beyond this point, except for the human expert who inspects the TBL changes on the original tagged corpus (ITC1). The corpus is automatically updated according to the accepted changes after the human expert's adjudication (table 11). The TBL model is retrained on the newly corrected corpus, and is thus updated after each iteration. The TBL deployed in this process is transformation-based learning on the fast-lane (fnTBL) by Ngai and Florian (2001), with the provided 40 rule templates at a threshold of 2. The output template for inspection is of the form P A B C, where P is the marked position (i), A is TBL changed tag (w_i/t^1), B is the current tag (w_i/t), and C is i's contextual information ($w_{i-2}/t\ w_{i-1}/t\ w_i/t\ w_{i+1}/t\ w_{i+2}/t$). See table 11 for sample results.

Finally, we performed manual error check on ITC2. Firstly, all tokens in ITC2 with POS-tags that are not in the revised tagset were checked and changed. This is done through building a tagset dictionary and passing ITC2 through it. Secondly, the TBL propagation process correctly reclassified some tokens in ITC1 with their new POS-tags introduced in the revised tagset. However, because of the small amount of corpus size used for TBL training, TBL lacked the capacity to apply learned rules widely on the ITC2 missing some instances that suppose to get the new POS-tags. To correct this, we used set of these new POS-tags to find tokens in ITC1 where they occurred, then we used these tokens to track all it's occurrences and their contexts for easy classification. This process corrected 4,994 *w/t* samples in ITC2 giving ITC3-current/first version of Igbo tagged corpus. Few examples of this process are shown in table 8. *ntachi obi* is an example of a multiword expression in Igbo meaning "steadfastness". They occur as a "link-pair" adjacent to each other without any intervening word. The second pair is complementing the meaning of the first. After TBL propagation method, as shown in ITC2 column, "ntachi" got a new POS-tag (NNCV) in 35 locations and it's pair "obi" also got NNCC in 35 locations. "obi" occurred 798 in entire text, it can occur on itself or adjacent to a verb or noun completing its meaning. We tracked all other locations in ITC2 where this link-pair occurred and inspect them to see whether they are suppose to get this tag or not. Outcome of our inspection is shown on the ITC3 column.

Token	Freq	ITC1	ITC2	ITC3
ntachi	38	NNC=35 VCO=1 NNAV=2	NNCV=35 NNC=1 NNAV=2	NNCV=38
obi	38	NNC=37 PRN = 1	NNCC=35 NNC=2 PRN=1	NNCC=38
ntụkwasị	67	VSI_XS=5 NNAV=1 VCO=6 NNC=55	NNCV=26 NNC=40 NNAV=1	NNCV=67
obi	67	NNC=67	NNCC=27 NNC=40	NNCC=67

Table 8: Some examples of manual error check and corrections

6 Evaluations

We present evaluation results for all the outputs of the above process: ITC0, ITC1, ITC2 and ITC3 to show improvement rates. For the evaluation performance, we split the corpora into 10 folds. 10-fold subsets were created by slicing the the corpora into 822 sentences, each is 25,981 words on the average. Slicing on the sentences is making sure that each piece contained full sentences (rather than cutting off the text in the middle of a sentence). For 10-fold steps and on closed vocabulary, we trained TBL classifier on 9-fold and

tested on the held-out. The results are summarised in table 9.

Fold	Accuracy			
	ITC0	ITC1	ITC2	ITC3
0	84.509	88.748	94.027	94.462
1	90.522	91.413	93.171	93.653
2	90.743	90.809	92.871	93.682
3	92.153	92.474	94.214	94.489
4	92.098	93.119	94.687	94.816
5	81.980	85.974	93.151	93.492
6	89.342	90.589	93.215	93.809
7	85.684	88.433	93.287	93.691
8	88.186	89.913	93.621	94.063
9	86.996	90.190	93.409	93.920
Average	**88.221**	**90.166**	**93.565**	**94.007**

Table 9: Simple accuracy on 10-fold evaluation

7 Discussion and Re-usability

We trained TBL classifiers on the inter-annotation agreement (IAA) annotated texts ($tl1, tl2, tl3, tl4, tl5$) with the assumption that errors flagged with the rule-based model generated will be the type of errors that occur in these texts. If we presume that these errors are evenly distributed, then we can assume that the most common types of errors will also occur frequently in the annotated texts, and are likely to be flagged in the full text. The effect of this assumption explored in section 5 is seen in table 10. A few samples from this experiment are displayed in table 11. The columns show the affected samples, TBL suggested tags, accepted (whether the TBL suggested tag was accepted by the human expert), manual correction (if TBL suggested tag and current ITC1 tag were wrong), and final state of tags. Interestingly, some tokens were correctly reclassified, even new tags introduced in the IAA exercises as a result of the tagset revision are correctly inserted into the main text. The Igbo corpus size of 263,854 tokens, which initially had 54 tags annotated according to the tagset reported in Onyenwe et al., (2014), now contains 66 tags, including all changes in the revised tagset.

We performed evaluation on the outputs from all of the process starting from the initial state of the main text to the improved state (ITC0 to ITC3) in section 6. From the table 9, we can deduce that there is constant improvement on the pattern consistency in the tagged corpus after each process. A total improvement score of 5.79% was achieved; manual cleaning gave 1.95% improvement, TBL propagation gave an additional 3.40%, and manual check up another 0.44%. Improvement processes flagged 62,385 word–tag pair positions which were inspected by an expert human annotator, contributed 23.93% improvement on the tagged Igbo corpus.

Token	Frequency of word in Maintxt	Frequency of word affected by the process
n’	11570	164
ndị	5755	3688
unu	3816	1389
a	3696	1350
onwe	831	828
banyere	611	503
olee	159	53
keenụ	3	1

Table 10: Frequency of words found in main text and TBL flagged samples

The Igbo language has 30 dialects as a result of nasality and aspiration[5]. Our tagset and corpus annotation is based on the standard Igbo, which omits the nasality and aspiration found in those dialects. The tagset and associated guideline are applicable to all 30 dialects, since these dialectal words play the same grammatical role as found in the standard Igbo texts, through which the tagset was developed. For example, the interrogative sentence *olee aha gị*? “what is your name?” in standard Igbo is said in different dialects as *ndee afụa gụ*?, *ndee awa ghụ*?, etc. “ndee” is equivalent to “olee” which makes the sentence interrogative, *afụa, ewa* is equal to “aha” and *gụ, ghụ* is equal to “gị”. Therefore, if we create a dictionary of word-types from the Bible in all dialects, with standard Igbo as a reference point, the annotated Bible corpus in standard Igbo can be used to annotate other dialects with minimal errors.

8 Conclusion and Further Work

We have presented a methodology to propagate POS-tag changes made during an inter-annotation agreement exercise due to tagset revisions on the main corpus. Our semi-automatic method, shows that even the new tags introduced in the IAA were found, and wrongly tagged tokens on ITC0 that were corrected in the IAA exercise were identified in the refined Igbo tagged corpus (ITC3). This is because the errors that TBL flagged are the types of errors that occur in the inter-annotation text. Through this process, we improved the quality of original Igbo tagged corpus by reflecting

[5] `http://www.ethnogue.com/language/ibo` [August, 2015]

Instance	TBL POS-tag	Accepted	Manual Change	Final POS-Tag	Meaning
ahụ/DEM	VPP	YES		VPP	see
ahụ/DEM	VPP	NO		DEM	that
n'/VAX	PREP	YES		PREP	in/on/from
na/VAX	CJN	YES		CJN	and
na-/NNC	VAX	YES		YES	auxiliary verb (AV)
onye/NNM	NNC	YES		YES	person
ndị/NNC	NNM	YES		YES	people of
onwe/PRNREF	PRNEMP	YES		PRNEMP	self
ya/PRN	PRNREF	NO		PRN	her/him
unu/NNM	PRN	YES		PRN	plural you
dịkwa/VCO	VSI_XS	YES		VSI_XS	is also
kọrọ/VrV	VPP_XS	NO		VrV	told
nyere/VCO	VSI_XS	NO	VrV	VrV	gave
ná/CJN	PREP	YES		PREP	in/on/from
a/DEM	PRN	NO		DEM	this
a/DEM	PRN	YES		PRN	impersonal pronoun (IP)
ana/VPP	VAX_BPRN	YES		VAX_BPRN	AV "na" with pronoun prefix "a"
m/PRN	BPRN	YES		BPRN	"I" bound to "a/e" pronoun
óké/NNC	NNH	YES		NNH	boundary
nwere/VrV	VMOV	YES		VMOV	[nwere ike] can
ike/NNC	VMOC	YES		VMOC	[nwere ike] can
ekwesị/VPP_XS	VPP	NO	BCN	BCN	right/correct
ònye/WH	NNC	NO		WH	who
ntachi/NNC	NNCV	YES		NNCV	[ntachi obi] steadfastness
obi/NNC	NNCC	YES		NNCC	[ntachi obi] steadfastness
esi/VPP	VSI_BPRN_XS	NO	VSI_BPRN	VSI_BPRN	simple verb "si" with pronoun prefix "e'

Table 11: Some samples of flagged locations inspected.

changes from the tagset revision made in the inter-annotation agreement exercise on it. We also applied TBL on each annotated text of the inter-annotation agreement exercise. These different rule sets generated can be used to identify locations for inspection across the whole corpus, for example, where the rules for most annotators suggest a tag where another annotator disagree. This finds and inspect where one annotator disagrees with majority, because among annotators, some are have better insight than others on a particular tag. Further more, manual error check was used to find and correct instances our propagation method affected but could not fire in all locations where they occurred. The evaluation result shows that we achieved an improvement of 5.786% over the entire process. The effort, time and money that would had been used to manually execute this were saved. In total, the entire processes gave 62,385 (23.92% of main corpus) positions inspected on the main corpus with 35,743 effective changes made.

The TBL propagation method used here can generalize to many annotation problems, especially low-resource languages since TBL has been classified to work well not only on large sized corpus but also on small amount of corpus. In Africa, of around 2000 languages in the continent, only a small number have featured in the NLP research field. This work is a good direction for them to co-opt our technique in POS-tagging their texts, which is a primary step in developing NLP resource tools.

The text of this annotated corpus is in standard Igbo. It is potentially re-usable on other dialects or genres towards developing annotated corpora with correctable errors. The only foreseen challenge in moving from religious genre used in this paper to other genres or from standard dialect to other dialects is the problem of unknown words, which is mainly caused by agglutinative nature of the language. We plan to further this research by developing the first Igbo POS-tagger, deal with handling of unknown words and develop annotated corpora for other dialects through the already tagged corpus. This work, to the best of our knowledge, developed the first tagged corpus for Igbo which is geared towards supporting computational NLP research on the language.

Acknowledgments

We acknowledge the financial support of Tertiary Education Trust Fund (TETFund), Nigeria and Nnamdi Azikiwe University, Awka, Nigeria.

References

Ron Artstein and Poesio Massimo. 2008. *Inter-coder agreement for computational linguistics.* MIT Press, (34)4:555–596.

Eric Brill. 1992. *A Simple Rule-based Part of Speech Tagger*. Proceedings of the Third Conference on Applied Natural Language Processing, ANLC '92. Association for Computational Linguistics, Stroudsburg, PA, USA.

Nọlue E. Emenanjo. 1999. *Elements of Modern Igbo Grammar: A Descriptive Approach*. Ibadan Oxford University Press.

Annette M. Green. 1997. *Kappa Statistics for Multiple Raters Using Categorical Classifications* Proceedings of the Twenty-Second Annual SAS Users Group International Conference, San, Diego, CA.

Sigrún Helgadóttir and Hrafn Loftsson and Eiríkur Rgnvaldsson. 2012. *Correcting Errors in a New Gold Standard for Tagging Icelandic Text.* LREC'14: 2944-2948.

Clara Ikekeonwu. 1999. *Igbo", Handbook of the International Phonetic Association*. Cambridge University Press.

Klaus Krippendorff. 1980. *Content Analysis: An Introduction to Its Methodology* Sage, Beverly Hills, CA.

Richard J. Landis and Gary G. Koch. 1977. *The measurement of observer agreement for categorical data* biometrics, 159–174, JSTOR.

Hrafn Loftsson 2009. *Correcting a POS-Tagged Corpus Using Three Complementary Methods* In Proceedings of EACL-09, 523–531.

Grace Ngai and Radu Florian. 2001. *Transformation-based learning in the fast lane*. In Proceedings of North Americal ACL.

Ikechukwu E. Onyenwe and Chinedu Uchechukwu and Mark Hepple. 2014. *Part-of-speech Tagset and Corpus Development for Igbo, an African.* LAW VIII, 2014.

James Pustejovsky and Amber Stubbs. 2012. *Natural Language Annotation for Machine Learning*. O'Reilly Media, Inc.

Fernández Raquel. 2011. *Assessing the Reliability of an Annotation Scheme for Indefinites Measuring Inter–annotator Agreement* Institute for Logic, Language and Computation University of Amsterdam.

Chinedu Uchechukwu. 2008. *African Language Data Processing: The Example of the Igbo Language*. 10th International pragmatics conference, Data processing in African languages.