LONTAR KOMPUTER VOL. 16, NO. 1 APRIL 2025 **p-ISSN 2088-1541**
DOI : 10.24843/LKJITI.2025.v16.i01.p04 **e-ISSN 2541-5832**
Accredited Sinta 2 by RISTEKDIKTI Decree No. 158/E/KPT/2021

Annotation Error Detection and Correction for Indonesian POS Tagging Corpus

Muhammad Alfian[a], Umi Laili Yuhana[a], Daniel Siahaan[a], Harum Munazharoh[b]

[a]Department of Informatics, Institut Teknologi Sepuluh Nopember
[b]Department of Indonesian Language and Literature, Universitas Airlangga
Jl. Raya ITS, Surabaya
[1]7025221023@student.its.ac.id
[2]yuhana@its.ac.id (Corresponding author)
[3]do.siahaan@its.ac.id
[4]harum.munazharoh@fib.unair.ac.id

Abstract

Linguistic Corpus is the primary material for training and evaluating machine learning models, especially for POS Tagging. However, the human-annotated corpus is not free from annotation errors. Annotation errors have a negative impact on model performance. Therefore, we propose annotation error detection and correction. We detect annotation errors in the Indonesian POS Tagging corpus using the n-gram variation method. Then, we correct the corpus using an expert-voting approach. Annotation error detection successfully collected 6,536 annotation error candidates. Each candidate has two possibilities: (i) an ambiguous word or (ii) an incorrect annotation. Annotation error correction validated and corrected the candidates using the majority-voting method in an expert group. Annotation error correction successfully identified and corrected 503 words from 1918 sentences. Then, we compared the performance of the POS Tagging model with the corpus before and after correction. The results showed a significant improvement in the F1-score value (+9.69%) compared to the uncorrected corpus.

Keywords: *Annotation Error Detection, Annotation Error Correction, POS Tagging*

1. Introduction

Linguistic Corpus is the most crucial element in NLP tasks such as NER [1], POS Tagging [2], etc. One of the essential roles of a corpus is as the primary material for training and evaluating machine learning-based models [3]. Humans usually annotate a Corpus for POS Tagging tasks [4], so they are prone to errors [5]. Annotation errors can occur in all corpora, even in gold standard corpora, corpora that are often considered error-free [3]. One example, annotation errors were found in the CONLL-2003 corpus, a corpus commonly used in NLP tasks in English [6].
Meanwhile, Indonesian is still developing a corpus for basic NLP tasks [7], [8]. Several researchers have developed a corpus for POS Tagging, including Lim [9] and Fu [10]. Among these corpora, Fu's corpus [10] Is the corpus with the largest number of words [11]. The Indonesian corpus is also not free from annotation errors. One example of an annotation error is shown in Table 1. The word *sekitar* (around) appears in several sentences but has different labels. Annotation errors and inconsistencies have a negative impact on model performance [3]. High-quality data is needed in the machine learning process [12]. Several researchers have attempted to improve the corpus with various techniques. One is Loftsson [13], using two stages to overcome annotation errors. In the first stage, the researcher used three methods to detect POS Tagging errors in the Icelandic corpus manually. The three methods are ngram, vote from 5 tagger tools, and shallow parsing. Based on the corrected corpus, the researcher re-evaluated and re-trained two POS Taggers for Icelandic in the second stage. The results of the second stage clearly show that the quality of PoS annotation in the IFD corpus significantly affects the accuracy of the tagger. Dickinson [14] proposed automatically correcting POS Tagging annotation errors in a corpus by adapting existing technology for POS Tagging disambiguation. The study was divided into two stages. The first stage is error detection. The researchers used an n-gram variation approach that identifies words that appear more than once with different labels as annotation error candidates. Then, in the second stage, the researchers adapted the taggers to account for the problematic

tag differences in the data. The use of ambiguity tags was shown to reduce the error rate of the corpus. Differences in tagging models had a greater impact on the accuracy of the correction.

Table 1. Example of annotation error

Sentence	Word Class
*Sejak kawasan **sekitar** sendang dijadikan sawah, air di sendang berkurang drastis.* (Since the area **around** the spring was turned into rice fields, the water in the spring has decreased drastically.)	RB
*Pohon besar di **sekitar** kandang terlihat rubuh.* (Large trees **around** the enclosure were seen to have collapsed.)	NN
*Dia menyelesaikan pembacaan pandangan FPDIP **sekitar** pukul 12.07 WIB* (He finished reading the FPDIP's views at **around** 12.07 WIB.)	IN

Angle [15] Detected errors in the Hindi POS Tagging corpus using an ensemble model of three POS Tag Models. Based on the predictions made by the three models, annotation errors were detected from the differences in the given tags. The researchers used three models that can predict POS tags accurately: the Hidden Markov Model, Support Vector Machine, Conditional Random Fields, and Logistic Regression. The ensemble model was built using a Fully Connected Neural Network. The Error Detection and Correction Model was built by training the Neural Network on the results of each model and its prediction probabilities. This approach has achieved an accuracy of 94.02% and can accurately identify most of the errors present in the corpus. This reduces the human effort required to clean the data to a minimum.

Several automation methods for annotation error detection and correction offer the effectiveness of models without human involvement [16]. Utilizing machines as a correction aid helps ensure data consistency. However, involving machines in annotation error correction does not guarantee data quality (validity). This is because humans and machines have differences in understanding Language [17]. Annotation error correction still requires human intervention to ensure the resulting data is consistent and valid in context.

Therefore, this study utilizes an expert-voting approach to correct the Indonesian POS Tagging corpus. The annotation errors candidates are distributed to several groups of at least three experts with an expert-voting approach. Each person is tasked with determining the correct word class of the given word. The proper word class is selected using the majority voting [18] approach of each group. This study does not propose a new method for annotation error correction. However, this study uses a technique in other cases to be applied in annotation error correction of the Indonesian POS Tagging corpus. Meanwhile, this study detects annotator errors with the n-gram approach [14]. The N-gram method detects annotation error candidates based on word frequency and label variations. The corpus used in this study is the Fu corpus [10]. After the corpus is corrected, we test the corrected corpus with the POS Tagging model using the Conditional Random Field (CRF) method [19].

Table 2. List of Word Classes in the Fu Corpus [10]

Category	Code	Word Class	Word List
Noun	NN	Common Noun	*buku, pipi, rupiah, km, sekarang*
	NNP	Proper Noun	*Indonesia, MH370, Li Li, SBY*
	SP	Subject-predicate structure	*katanya, sebutnya, tuturnya*
Pronoun	PRD	Demonstrative Pronoun	*ini, itu, sini, sana, tersebut*
	PRF	Reflexive Pronoun	*sendiri, diri, dirinya*
	PRI	Indefinite Pronoun	*siapapun, apapun, seseorang*
	PRL	Relative Pronoun	*yang*
	PRP	Personal Pronoun	*saya, kamu, dia, kami, kalian*

	WH	Question	*apa, siapa, mana, bagaimana*
Adjective	JJ	Adjective	*besar, tinggi, manis, cerdik*
	JJS	Adjective, superlative	*terdekat, terbesar, terpenting*
Verbs	VB	Verb	*ada, melihat, gagal, menyoroti*
	VO	Verb-object structure	*meningkatnya, terbentuknya*
Adverbs	MD	Auxilary Verb	*harus, perlu, boleh, adalah, mau*
	RB	Adverb	*sudah, tidak, sangat, juga*
Conjunction	CC	Coordinating Conjunction	*dan, tetapi, atau*
	SC	Subordinating Conjunction	*kalau, jika, sementara itu*
Interjection	IN	Preposition	*di, ke, oleh, untuk, dari, antara*
	PO	Preposition Object Structure	*untuknya, antaranya, olehku*
Determiner	UH	Interjection	*oh, hai, ya, sih, mari*
	DT	Determiner	*para, sang, si*
	CD	Cardinal Number	*satu, dua, 79, 2017, 0.1, ratus*
	OD	Ordinal Number	*pertama, ketiga, ke-6*
	ID	Indefinite Number	*puluhan, 30-an, beberapa*
Particle	P	Particle	*pun, -lah, -kah*
Symbols	SYM	Symbol	*+, %, @, $, 15/2/2017, 13:00, Rp*
	Z	Punctuation	*“,.?”()*
Miscellaneous	FW	Foreign Word	*poetry, technology, out, world*
	X	Unknown	*yagg, busaway, saaat*

2. Research Methods

The design of the annotation error detection and correction system is shown in Figure 1. In the first step, we group sentences into several groups based on the completeness and arrangement of their elements. The Fu corpus is grouped based on the type of sentence using the sentence element identification approach. Grouping sentences is the selection of sentences to make the correction process effective. The group of sentences is selected using predetermined exclusion criteria. The selected sentences are parsed into words and stored as a filtered corpus. Then, the words are processed using n-gram variation to identify annotation error candidates. N-gram variation collects words that appear more than once and have different labels as mislabeling candidates. The candidates are distributed to several expert groups (EG) for validation and correction. Each expert group is responsible for the consistency and validation of the data. The results of the corpus repair are returned to the filtered corpus and then tested on the POS Tagging program.

We compare the model's performance using the filtered corpus before and after correction. We also compare the dataset ratio to measure the model's reliability in a limited training data scenario. The performance of both models was assessed using the Macro F1-Score matrix to avoid majority label bias.

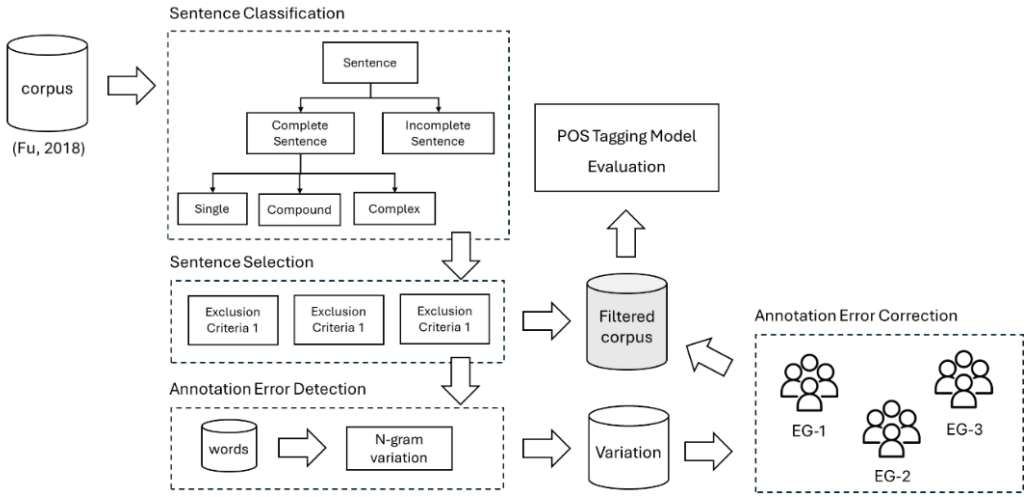

Figure 1. Annotation Error Detection and Correction System Diagram

2.1. Corpus

This study uses a corpus created by Fu & Lin [10]. The corpus consists of 21,024 sentences with 355,010 words. This corpus uses 29 standard tagsets created by Fu & Lin [10]. The tagsets are shown in Table 2. The tagsets are grouped into 11 categories. The category with the most word classes is the Pronoun category. This corpus was chosen because it contains the largest number of words among previous Indonesian corpora [11].

2.2. Sentence Classification

Before validation, the corpus was first grouped based on the type of sentence. Based on the completeness of its elements, sentences in Indonesian are divided into complete sentences and incomplete sentences. A complete sentence is a sentence that has at least one subject and one predicate. At the end of this subsection, we discuss removing sentence elements, including subjects and predicates. Meanwhile, an incomplete sentence is a sentence that does not meet these criteria. Based on the composition of its elements, complete sentences are divided into two types: Single and compound sentences [20]. We also added another type that is more complex, namely compound-complex sentences. A Single Sentence is a sentence that has at least one subject and predicate. Meanwhile, a compound sentence is a sentence that has more than one subject and predicate without a conjunction. A compound-complex sentence is a compound sentence that contains a conjunction. The list of conjunctions [21] can be seen in Table 3. The element identification process utilizes the dependency parsing program.

Table 3. List of Conjunctions

Category	Word List
Coordinative Conjunction	*Dan, atau, melainkan, padahal, sedangkan, serta, tetapi, tapi, dan/atau*
Subordinative Conjunction	*sejak, sedari, semenjak, begitu, demi, ketika, sambil, selagi, selama, sementara, seraya, tatkala, sewaktu, setelah, sebelum, sesudah, sehabis, selesai, seusai, hingga, sampai, apabila, jika, jikalau, kalau, manakala, andaikan, seandainya, sekiranya, seumpamanya, andai kata, agar, biar, supaya, biarpun, meski, meskipun, sekalipun, kendati, kendatipun, sungguhpun, walau, walaupun, alih-alih, daripada, ibarat, laksana, seakan-akan, sebagai, sebagaimana, seolah-olah, seperti, karena, sebab, oleh karena, oleh sebab, maka, makanya, sehingga, sampai, sampai-sampai, dengan, tanpa, bahwa, yang, Biarpun demikian, sekalipun demikian, walaupun begitu, Kemudian, sesudah itu, selanjutnya, Tambahan pula, lagi pula, selain itu, Sebaliknya, Sesungguhnya, sebenarnya, Bahkan, malah,*

	malahan, Akan tetapi, namun, Kecuali itu, di samping itu, Oleh karena itu, oleh sebab itu, Sebelum itu, maupun, entah

This program analyzes the relationship between one word and another in a sentence. We use the Stanza library to create a sentence element identification program. We identify five sentence elements: subject, predicate, object, description, and complement. We upload the program to the GitHub platform so other researchers can use it. Figure 2 explains the pseudocode for identifying sentence elements. Based on syntactic theory, we map the characteristics of these elements based on dependency labels and word classes. Table 4 shows the criteria we use to categorize sentence elements. If a word has a dependency class as its root and its POS Tag is verb/noun/adjective/numeric, then the word is a predicate element in the sentence. Meanwhile, if a word has a dependency class obl, then the word is an adverbial element, except for transitive predicates, where obl indicates a complement element. One word can occupy more than one element.

Table 4. Sentence Element Criteria

Category	POS	Dependency
Subject	-	nsubj, nsubj:pass
Predicate	VERB, NOUN, ADJ, NUM	root, cop, attr, acl, advmod
Object	-	obj, iobj
Adverbial	-	advmod, npadvmod, obl
Complement	-	xcomp, ccomp, acomp, obl*

2.3. Sentence Selection

We select sentences before validation by experts. Sentences are chosen by experts based on the level of ease of analysis. The following are the criteria compiled for sentence selection.

E1. Sentences are not dialogue sentences
E2. Sentences consist of only one sentence, no more than one sentence (multiple sentences).
E3. Sentences that are validated are simple sentences (single sentences)

```
FOR w ← S
  Pred_T = False
  IF d(w) ϵ D_S THEN Sub = w
  IF d(w) ϵ D_P AND p(w) ϵ P_P THEN
    Pred = w
    FOR c ← S
      IF d(c) = obj AND d(w) = root THEN
        Pred_T = True
  IF d(w) ϵ D_A THEN Adv = w
          IF Pred_T = False AND d(w) ϵ D_C THEN
    Comp = w
  ELSE IF Pred_T = True AND d(w) = obl THEN
    Comp = w
  RETURN Sub, Pred, Adv, Comp
```

Figure 2. Pseudocode for Sentence Element Identification

2.4. Annotation Error Detection

Annotation Error Detection is an effort to identify potential annotation errors to improve the quality of the corpus. Annotation Error Detection is usually performed on the linguistic corpus, especially POS Tagging [3]. We detect annotation errors in each word using the n-gram variation method. The n-gram variation method is straightforward. This method calculates the frequency of words in the corpus and then filters out words that appear more than once. Then, each filtered word is calculated for its word class label variation. For example, as shown in Table 1, the word "around" appears in three sentences and has word classes RB, NN, and IN. So, the label variation is three.

The word is identified as an annotation error candidate if there is more than one label. We use the pandas library to create this method. The annotation error candidates have two possibilities: i) ambiguity: the word has several word class options depending on the context, or ii) annotation error: inconsistency or annotation error. The more similar the candidate contexts between sentences, the more likely the word is detected as an annotation error. The Annotation Error Correction processes the annotation error candidate to determine the possibility of the two possibilities.

2.5. Annotation Error Correction

Annotation Error Correction is the stage of validation and correction of annotation errors. We use an expert-voting approach involving language experts divided into groups of three to four members. Experts must have minimum competencies, such as understanding the morphology and syntax of the Indonesian language. Experts must be observant in determining word classes based on the context of the words. Experts also have the right to eliminate the use of words in sentences that are not commonly used (outliers) [22]. Each expert group is responsible for ensuring the consistency and validation of the words. To determine the word class, we adopted majority voting [23]. Group members discuss the appropriate label for a particular word. If there is a difference of opinion, a vote is taken to determine the proper label. The label with the most votes becomes the label for the word.

2.6. Model Evaluation

We evaluate the performance of the POS Tagging model with two types of corpora. The corpora are a filtered but uncorrected corpus and a filtered and corrected corpus. This evaluation is carried out with five different testing data ratio schemes: the ratios 10:90, 20:80, 30:70, 40:60, and 50:50. The testing uses the Conditional Random Field (CRF) [24] method with the Macro F1-Score ($aF1$) metric as its reference [25]. Macro F1-Score (2) obtained from the average of F1-Score values (1). The purpose of evaluating the POS Tagging model is to compare the model's performance before and after annotation error correction.

$$F1 = \frac{TP}{TP + \frac{1}{2}(FP + FN)} \quad (1)$$

$$aF1 = \frac{\sum_{i=1}^{n} F1_i}{n} \quad (2)$$

3. Results and Discussion

The explanation of the results of our experiments and tests is arranged according to the sequence in the system diagram. Each process is explained in several sub-chapters, including sentence classification, sentence selection, annotation error detection, annotation error correction, and model evaluation.

3.1. Sentence Classification

We use the sentence element recognition program explained in the previous chapter to classify sentences. The program uses the Python language by utilizing the Stanza library. The sentence classification results are shown in Figure 3. The graph shows that about 15% of sentences in the Fu corpus are incomplete sentences. These sentences do not have either a subject or a predicate. Incomplete sentences usually have fewer words than complete sentences. Examples of complete and incomplete sentences are shown in Table 5. Complete sentences are easier to analyze because each word has clear grammar in a sentence.

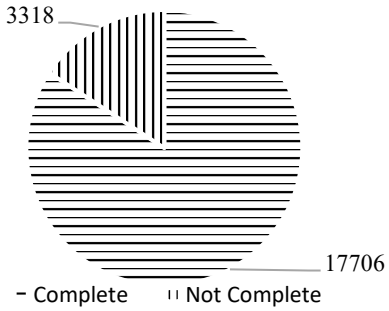

Figure 3. Sentence Classification Results based on the completeness of its elements

Table 5. Examples of Complete and Incomplete Sentences

Category	Sentence
Complete Sentence	*Buku tebal-tebal pun dibaca oleh anak itu juga.* (The child also read thick books.)
Not a Complete Sentence	*Nomor kosong.* (Empty number.)

The 17,706 complete sentences are grouped again based on their element composition. Figure 4 shows that the Fu corpus has more compound-complex sentences than other types of sentences. About 51% of the sentences are compound-complex sentences. The rest are compound sentences (28%) and single sentences (20%). Table 6 shows examples of single, compound, and compound-complex sentences. At a glance, compound-complex sentences have more words than other types.

3.2. Sentence Selection

We selected sentences based on the criteria that were set at the beginning. The first criterion is that the sentence is not a dialogue sentence. We conducted manual observation to identify dialogue sentences. Table 7 shows an example of a dialogue sentence. The sentence has the most striking characteristic, namely the use of double quotation marks. We used the search feature to search for sentences containing double quotation marks and reread each sentence to select dialogue sentences. Then the second criterion is that the sentence consists of only one sentence. Some sentences in the Fu corpus sometimes contain more than one sentence. Examples of such double sentences are shown in Table 7. We selected by identifying punctuation marks that are often used to end sentences, such as periods (.), exclamation marks (!), and questions (?). Then we reread sentences that contain more than one punctuation mark. And the third criterion is that we only use sentences with a simple composition of elements, namely, single sentences, for validation. The results of the sentence selection are shown in Table 8. In the first stage (E1), we eliminated dialogue sentences from each sentence category. Then, in the second stage (E2), we eliminated multiple sentences. And in the last stage (E3), we selected sentences with the single sentence category as sentences to be validated. So, after the sentence selection stage, the total corpus was 3,055 sentences.

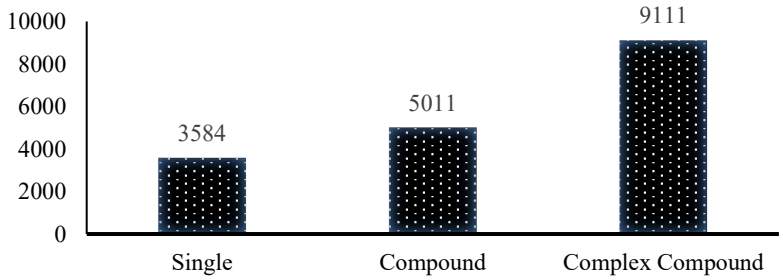

Figure 4. Sentence Classification Results based on the content of its elements

Table 6. Examples of Simple, Compound, and Complex Compound Sentences

Category	Sentence
Single Sentence	*Kampus kami terletak di kaki gunung.* (Our campus is located at the foot of the mountain.)
Compound Sentence	*Rosa disidang terkait kasus suap pembangunan wisma Atlet Sea Games.* (Rosa was tried in connection with the bribery case for the construction of the Sea Games Athletes' Village.)
Complex Compound Sentence	*Namun tidak banyak orang yang tahu bahwa musiknya sangat dipengaruhi oleh masa kecilnya di Tai O, sebuah desa nelayan kecil yang terletak di sisi barat Pulau Lantau.* (But not many people know that his childhood heavily influences his music in Tai O, a small fishing village located on the west side of Lantau Island.)

Table 7. Examples of Dialogue Sentences and Multiple Sentences

Category	Sentence
Dialogue Sentence	*"Kita sterilisasi dan kita lepas lagi," ujar Kusdiana.* ("We sterilize and rerelease them," said Kusdiana.)
Double Sentence	*Tiket masuk, dipatok Rp 20.000. Detikcom berjalan menyusuri setiap kandang.* (Entrance tickets are priced at Rp 20,000. Detikcom walked through each cage.)

Table 8. Sentence Selection Results based on Exclusion Criteria

Category	Freq	E1	E2	E3
Single Sentence	3584	3084	3055	3055
Compound Sentence	5011	4109	4038	0
Complex Compound Sentence	9111	6613	6469	0
Total Sentence	**17706**	**13806**	**13562**	**3055**

3.3. Annotation Error Detection

After going through sentence selection, we parse the corpus into words. The total number of labeled words we have is 31,015. We use the n-gram variation method to identify annotation error candidates. This word search is done without considering case sensitivity. Table 9 shows the examples of Annotation Error Candidates. The word "*dia*" appears 175 times, but has various label variations (DT, PRL, PRP, and VB). Meanwhile, the word "*bentuk*" appears four times and has the same/consistent label (NN). Experts will correct the candidates by reading sentences containing these words. The total number of annotation error candidates successfully collected was 6,536 words.

Table 9. Examples of Annotation Error Candidates

Word	Freq	Word Class
Dia	175	DT, PRL, PRP, VB
Nanti	17	MD, NN, RB
Sekitar	67	ID, IN, NN, RB
Mobil	40	NN, VB
Kantor	27	NN, NNP

3.4. Annotation Error Correction

We involved 11 experts who had an understanding of morphology and syntax. We divided the experts into three groups with 3-4 members. A total of 6,536 annotation error candidates were divided into three parts in alphabetical order. Each group was responsible for validating an error candidate. Each candidate contains several sentences containing the word. Each expert was tasked with reading all the sentences in the annotation error candidate. If the expert found a word class that was not quite right, the expert was asked to correct it and determine the correct word class. If the word class was correct, the expert did not need to correct it. We used the majority

voting method from expert answers if there was disagreement between experts. Table 10 shows an example of correcting the word "itu". The word "itu" has the initial word classes "PRD" and "DT". Then the expert analyzed and corrected the word class to "PRD".

After the expert finished correcting the words, we summarized the results. We found 503 words from 1918 sentences with annotation errors. This number equals 8% of the total annotation error candidates found. This shows that the rest (6,033 words) have more than one word-class option (ambiguous). We summarize the results of word-class correction based on the word classes. Table 11 shows a significant increase in NN, JJ, VB, IN, and DT word classes. Meanwhile, there was a significant reduction in word classes NNP, PRD, PRP, and MD. The revised word classes were then returned to the filtered corpus. After that, we conducted model evaluation on both types of corpora.

Table 10. Example of Correction of Word Class from *itu* (that/the)

Sentence	Initial	Revised
*Museum seni **itu** belum terbuka untuk umum.* (**The** art museum is not yet open to the public.)	PRD	PRD
*Lihat, **itu** siapa.* (Look, who's **that**?)	PRD	PRD
*Tindak tanduk orang **itu** diawasi polisi.* (**The** person's actions were monitored by the police.)	DT	PRD
*Akibat serangan angin **itu** 12 rumah warga mengalami rusak berat.* (As a result of **the** wind attack, 12 residents' houses were severely damaged.)	DT	PRD

Table 11. Word Selection Results based on Word Class

Category	Class	Initial	Revised	Difference
Noun	NN	7313	6888	425
	NNP	5119	5391	-272
	SP	7	12	-5
Pronoun	PRD	136	999	-863
	PRF	27	62	-35
	PRI	6	7	-1
	PRL	2	1	1
	PRP	769	901	-132
	WH	130	120	10
Adjective	JJ	963	839	124
	JJS	20	24	-4
Verbs	VB	4065	3863	202
	VO	29	30	-1
Adverbs	MD	399	570	-171
	RB	2042	2047	-5
Conjunction	CC	17	24	-7
	SC	103	176	-73
Interjection	IN	2265	2140	125
	PO	22	24	-2
Determiner	UH	108	175	-67
	DT	1083	156	927
	CD	718	749	-31
	OD	34	37	-3
	ID	174	210	-36
Particle	P	94	114	-20
Symbols	SYM	198	197	1
	Z	4978	4976	2
Miscellaneous	FW	113	112	1
	X	13	11	2

3.5. Model Evaluation

We evaluated the model using two types of corpora: the unrevised filtered corpus (initial) and the revised corpus, which consists of 31,015 words. We conducted the test with different test ratios.

Table 12 shows the results of the POS Tagging model evaluation from the test scenario. The table shows an improvement in the F1-score value from the unrevised corpus to the revised corpus. The increase in the number of F1-scores is quite significant (+9.69%), with the highest increase in a test ratio of 40:60. This increase is comparable to the number of words successfully corrected by experts (8%). Thus, this increase occurred due to reduced annotation errors in the previous corpus. It shows that corpus improvement can significantly improve POS Tagging performance.

Table 12. F1-score Evaluation of POS Tagging Model

Test Ratio	Initial	Revised	Improvement
10:90	76.66%	84.96%	+8.30%
20:80	76.13%	83.05%	+6.92%
30:70	72.77%	79.97%	+7.20%
40:60	70.41%	80.1%	+9.69%
50:50	67.07%	72.29%	+5.22%

4. Conclusion

Corpus is the most important element in an NLP task to train and evaluate machine learning based models. High-quality data is needed to produce optimal model performance. However, a corpus for an NLP task, such as POS Tagging, can potentially have annotation errors. Annotation errors have a negative impact on model performance. Therefore, we propose annotation error detection and correction. We detect annotation errors in the Indonesian POS Tagging corpus using the n-gram variation method. Then, we correct the corpus using an expert-voting approach. The detected annotation errors are distributed to several expert groups and corrected using the majority voting technique. This study does not propose a new method for annotation error correction. However, this study uses a technique in other cases to be applied in annotation error detection and correction of the Indonesian POS Tagging corpus. Annotation error detection successfully collected 6,536 candidates. The candidates have two possibilities: (i) the word is ambiguous or (ii) annotated incorrectly. Then, the expert-voting process successfully identified 6,033 ambiguous words and corrected 503. The words have been corrected and validated as words with annotation errors. Then, we compared the performance of the POS Tagging model from the corpus before and after correction. The results showed a significant improvement in the F1-score value (+9.69%) compared to the uncorrected corpus.

Acknowledgement

The authors would like to thank the 25 students from the language and literature department of Airlangga University for their assistance in data analysis and discussions on Indonesian grammar. The authors would like to thank the Ministry of Education, Culture, Research, and Technology of the Republic of Indonesia and Institut Teknologi Sepuluh Nopember for the funding to support the collaborative research in this paper through the Doctoral Dissertation Research grant and Fresh-Graduate Scholarship. This research was supported in part by Institut Teknologi Sepuluh Nopember under Fresh-Graduated Scholarship, scholarship number 1333/IT2/T/HK.00.01/2022 and in part by the Ministry of Education, Culture, Research, and Technology of Indonesia under 2024 Doctoral Dissertation Research grant, grant number 038/E5/PG.02.00.PL/2024.

References

[1] D. Kim *et al.*, “A Neural Named Entity Recognition and Multi-Type Normalization Tool for Biomedical Text Mining,” *IEEE Access*, vol. 7, pp. 73729–73740, 2019, doi: 10.1109/ACCESS.2019.2920708.

[2] S. N. A. N. Ariffin and S. Tiun, “Improved POS Tagging Model for Malay Twitter Data based on Machine Learning Algorithm,” *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 7, pp. 229–234, 2022, doi: 10.14569/IJACSA.2022.0130730.

[3] J. C. Klie, B. Webber, and I. Gurevych, “Annotation Error Detection: Analyzing the Past and Present for a More Coherent Future,” *Computational Linguistics*, vol. 49, no. 1, pp. 157–198, 2023, doi: 10.1162/coli_a_00464.

[4] N. J. Dobbins, T. Mullen, Ö. Uzuner, and M. Yetisgen, "The Leaf Clinical Trials Corpus: a new resource for query generation from clinical trial eligibility criteria," *Sci Data*, vol. 9, no. 490, 2022, doi: 10.1038/s41597-022-01521-0.

[5] C. G. Northcutt, L. Jiang, and I. L. Chuang, "Confident learning: Estimating uncertainty in dataset labels," *Journal of Artificial Intelligence Research*, vol. 70, pp. 1373–1411, 2021, doi: 10.1613/JAIR.1.12125.

[6] S. A. A. Shah, M. Ali Masood, and A. Yasin, "Dark Web: E-Commerce Information Extraction Based on Name Entity Recognition Using Bidirectional-LSTM," *IEEE Access*, vol. 10, pp. 99633–99645, 2022, doi: 10.1109/ACCESS.2022.3206539.

[7] Y. Fu, N. Lin, X. Lin, and S. Jiang, "Towards corpus and model: Hierarchical structured-attention-based features for Indonesian named entity recognition," *Journal of Intelligent and Fuzzy Systems*, vol. 41, no. 1, pp. 563–574, 2021, doi: 10.3233/JIFS-202286.

[8] I. M. S. Putra, D. Siahaan, and A. Saikhu, "SNLI Indo: A recognizing textual entailment dataset in Indonesian derived from the Stanford Natural Language Inference dataset," *Data in Brief*, vol. 52, p. 109998, 2024, doi: 10.1016/j.dib.2023.109998.

[9] E. S. Lim *et al.*, "ICON: a linguistically-motivated large-scale benchmark Indonesian constituency treebank," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 8, pp. 1–34, Aug. 2023, doi: 10.1145/3609798.

[10] S. Fu, N. Lin, G. Zhu, and S. Jiang, "Towards Indonesian Part-of-Speech tagging: Corpus and models," *2018 International Conference on Asian Language Processing (IALP)*, vol. 1, pp. 303–307, 2018.

[11] M. Alfian, U. L. Yuhana, and D. Siahaan, "Indonesian Part-of-Speech tagger: A comparative study," in *2023 10th International Conference on Advanced Informatics: Concept, Theory and Application (ICAICTA)*, IEEE, Oct. 2023, pp. 1–6. doi: 10.1109/ICAICTA59291.2023.10390353.

[12] H. Song, M. Kim, D. Park, Y. Shin, and J. G. Lee, "Learning From Noisy Labels With Deep Neural Networks: A Survey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 11, pp. 8135–8153, 2023, doi: 10.1109/TNNLS.2022.3152527.

[13] P. Květoň and K. Oliva, "(Semi-)automatic detection of errors in PoS-tagged corpora," in *COLING '02: Proceedings of the 19th international conference on Computational linguistics*, 2002, pp. 1–7. doi: 10.3115/1072228.1072249.

[14] M. Dickinson, "From detecting errors to automatically correcting them," in *EACL 2006 - 11th Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference*, 2006.

[15] S. Angle, P. Mishra, and D. M. Sharma, "Automated error correction and validation for POS tagging of hindi," in *Proceedings of the 32nd Pacific Asia Conference on Language, Information and Computation, PACLIC 2018*, 2018.

[16] Y. Yanfi, R. Setiawan, H. Soeparno, and W. Budiharto, "SPECIL: spell error corpus for the Indonesian language," *IEEE Access*, vol. 11, pp. 93227–93237, 2023, doi: 10.1109/ACCESS.2023.3307712.

[17] M. Chen, "Trust, understanding, and machine translation: the task of translation and the responsibility of the translator," *AI & Soc*, vol. 39, pp. 2307–2319, 2023, doi: 10.1007/s00146-023-01681-6.

[18] Z. Chen, L. Jiang, and C. Li, "Label augmented and weighted majority voting for crowdsourcing," *Inf Sci (N Y)*, vol. 606, pp. 397–409, 2022, doi: 10.1016/j.ins.2022.05.066.

[19] S. Warjri, P. Pakray, S. A. Lyngdoh, and A. K. Maji, "Part-of-speech (POS) tagging using conditional random field (CRF) model for Khasi corpora," *International Journal of Speech Technology*, vol. 24, no. 4, pp. 853–864, 2021, doi: 10.1007/s10772-021-09860-w.

[20] A. Rahmawati, H. Setiawan, and F. Meliasanti, "Analisis Kalimat Tunggal dan Majemuk Pada Rubrik Pendidikan di kompas.com Serta Rekomendasinya Sebagai Bahan Ajar di SMP," *Jurnal Educatio*, vol. 7, no. 4, pp. 1602–1606, 2021.

[21] F. Yani, "The Comparison Between English Conjunction and Indonesian Conjunctiona," *Cendikia : Media Jurnal Ilmiah Pendidikan*, vol. 11, no. 2, pp. 71–81, 2021, doi: 10.35335/cendikia.v11i2.1667.

[22] B. A. Smitha and R. K. N. Praveen, "ORDSAENet: Outlier Resilient Semantic Featured Deep Driven Sentiment Analysis Model for Education Domain," *Journal of Machine and Computing*, vol. 3, no. 4, pp. 408–430, 2023, doi: 10.53759/7669/jmc202303034.

[23] T. Karadeniz, H. H. Maraş, G. Tokdemir, and H. Ergezer, "Two Majority Voting Classifiers Applied to Heart Disease Prediction," *Applied Sciences (Switzerland)*, vol. 13, no. 6, p. 3767, 2023, doi: 10.3390/app13063767.

[24] A. Pradhan and A. Yajnik, "Parts-of-Speech tagging of Nepali texts with Bidirectional LSTM, Conditional Random Fields and HMM," *Multimedia Tools and Applications*, vol. 83, pp. 9893–9909, Jun. 2023, doi: 10.1007/s11042-023-15679-1.

[25] A. Turchin, S. Masharsky, and M. Zitnik, "Comparison of BERT implementations for natural language processing of narrative medical documents," *Informatics in Medicine Unlocked*, vol. 36, p. 101139, 2023, doi: 10.1016/j.imu.2022.101139.