BILINGUAL PUBLISHING GROUP

Forum for Linguistic Studies

https://journals.bilpubgroup.com/index.php/fls

ARTICLE

A Gold Standard Dataset for Javanese Tokenization, POS Tagging, Morphological Feature Tagging, and Dependency Parsing

Ika Alfina [1*] **, *Arlisa Yuliawati*** [1] **, *Dipta Tanaya*** [1] **, *Arawinda Dinakaramani*** [1] **, *Daniel Zeman*** [2]

[1] *Faculty of Computer Science, Universitas Indonesia, Depok 16424, Indonesia*

[2] *Faculty of Mathematics and Physics, Charles University, Praha CZ-11800, Czechia*

ABSTRACT

Javanese, a regional language in Indonesia with more than 68 million speakers, is a low-resource language in the Natural Language Processing (NLP) field because it needs more language resources in both dataset and NLP tools. In this work, we developed a gold standard dataset of 1,000 sentences and 14,323 words for Javanese for four NLP tasks: tokenization, part-of-speech (POS) tagging, morphological feature tagging, and dependency parsing. This dataset is in the CoNLL-U format that conforms with the Universal Dependencies (UD) annotation guidelines. We involved native Javanese speakers as the annotators. Javanese sentences are taken from grammar books, Wikipedia, and online newspapers. We build models for tokenization, POS tagging, morphological feature tagging, and dependency parsing using UDPipe to evaluate the dataset's quality. The evaluation was conducted with the 10-fold cross-validation method. For the tokenization task, our model has an F1 score of 99.53%, 72.01%, 97.11%, and 95.90% for segmenting tokens, multiword tokens (MWT), syntactic words, and sentences, respectively. For POS and morphological feature tagging from gold tokenization, the model has an F1-score of 87.22% and 86.66% for POS tagging and morphological feature tagging. Finally, for the dependency parsing task, parsing from gold tokenization with gold tags has an Unlabeled Attachment Score (UAS) of 77.08% and a Labeled Attachment Score (LAS) of 71.21%.

Keywords: Annotation Guidelines; Dependency Parsing; Low-Resource Language; Morphological Feature Tagging; POS Tagging; Tokenization; Universal Dependencies

*CORRESPONDING AUTHOR:

Ika Alfina, Faculty of Computer Science, Universitas Indonesia, Depok 16424, Indonesia; Email: ika.alfina@cs.ui.ac.id

ARTICLE INFO

Received: 27 July 2024 | Revised: 16 August 2024 | Accepted: 19 August 2024 | Published Online: 5 November 2024
DOI: https://doi.org/10.30564/fls.v6i5.6957

CITATION

Alfina, I., Yuliawati, A., Tanaya, D., et al., 2024. A Gold Standard Dataset for Javanese Tokenization, POS Tagging, Morphological Feature Tagging, and Dependency Parsing. Forum for Linguistic Studies. 6(5): 131–148. DOI: https://doi.org/10.30564/fls.v6i5.6957

1. Introduction

Javanese is one of approximately 700 ethnic languages in Indonesia. It is spoken by more than 68 million people and ranked 28th in the world in terms of the number of speakers[1]. Most of the Javanese speakers live in Central Java and East Java. Javanese is also spoken in Suriname, Sri Lanka, and New Caledonia by immigrant communities there. Javanese belongs to the Austronesian language family, specifically the Malayo-Polynesian group. Meanwhile, Indonesian belongs to another group of Austronesian languages, the Malayo-Sumbawan[1].

Javanese has been traditionally written using Javanese script. However, for practical purposes, Javanese speakers nowadays rarely use Javanese script and switch to Latin script. In our work, we process Javanese text in Latin script. Therefore, research on Javanese Natural Language Processing (NLP) using Javanese script is outside the scope of our study.

Even though there is a huge number of Javanese speakers, the development of NLP research for Javanese is still concerning[2]. The five stages of analysis in processing natural language are defined: 1) tokenization, 2) lexicon analysis, 3) syntactic analysis, 4) semantic analysis, and 5) pragmatic analysis[3]. Unfortunately, so far, Javanese NLP research is still stagnant at the level of tokenization and lexical analysis.

In the area of word segmentation or tokenization, the Finite State Transducer (FST) was used to split Javanese words written in Latin script into syllables[4]. After that, a morphological segmentation task that split a word into stems and affixes was conducted[5].

Stemming and part-of-speech (POS) tagging tasks were conducted using lexical analysis. For stemming, a Javanese stemmer was built[6, 7], by adapting the stemming algorithm for Indonesian[8]. For POS tagging for Javanese, several works have tried to develop a Javanese POS tagger[9–12].

In 2016, a POS tagger was built for Javanese Krama[10], the formal Javanese. They proposed the Javanese POS tag set of 19 tags and built a manually annotated dataset. The paper does not state the exact size of the dataset. However, they mention two training datasets with sizes of 2,380 words (Dataset A) and 8,488 words (Dataset B), respectively, but did not inform the size of the test dataset. They built a POS tagger model using a rule-based and Maximum Entropy method with an accuracy of 97.67%[13].

Three works from Telkom University also developed POS taggers for Javanese[9, 10, 12]. All of them adopted the proposed Javanese POS tag set[10]. The first work used the Hidden Markov Model (HMM) to build a POS tagger for Java Ngoko[11, 14], the informal Javanese, with a reported accuracy of 92.6%. They built a dataset of 126 sentences and 1,770 words, and the original sentences were taken from online news in Javanese. The second work used the Support Vector Machine (SVM)[9, 15], with a reported accuracy of 77%. Unfortunately, there is no information about the dataset they used in this paper. Finally, the third work used the Conditional Random Field (CRF) to train the POS tagger model for Javanese Krama with an accuracy of 67%[12, 16]. They build a dataset of 3,000 words in which the original sentences are also taken from online news.

All previous works in building Javanese POS taggers did not share their datasets with the public[9–12], making it difficult for future researchers to compare model performance or make improvements. This scarcity of public Javanese NLP datasets motivated us to develop the first one for some basic tasks in NLP. For the dataset we built to synergize with the resources of the high-resource language, we decided to make the dataset using a format or framework that is commonly used by other languages in the world, and our choice fell on Universal Dependencies (UD)[17].

UD is a framework for annotating the grammatical structure of natural language sentences. It is a cross-linguistically consistent scheme for syntactic annotation. However, since each language has a specific structure, we need to propose some adjustments so that certain languages comply with the UD annotation guidelines.

A UD dataset/treebank represents annotations for many tasks in morphology and syntax: 1) tokenization/word segmentation, 2) lemmatization, 3) POS tagging, 4) morphological feature tagging, and 5) dependency parsing. Due to limited resources, our work only annotated our dataset for four of five tasks, excluding annotation for the lemmatization task. As far as we know, our dataset is the first for morphological feature tagging and dependency parsing tasks for Javanese.

Furthermore, to evaluate the quality of the resulting dataset, we built NLP models for tokenization, POS tagging, morphological features tagging, and dependency parsing using UDPipe[18, 19]. UDPipe is a trainable pipeline that per-

forms tokenization, morphological analysis, POS tagging, lemmatization, and dependency parsing. The contributions of our work are three-fold:

1. We proposed the annotation guidelines to annotate Javanese text for the UD dataset that covers four tasks: tokenization, POS tagging, morphological feature tagging, and dependency parsing;
2. Using the proposed annotation guidelines, we constructed the first publicly available Javanese dataset, consisting of 1,000 sentences and 14,323 words for the four tasks. This treebank has been a part of the UD dataset since UD v2.9, with the latest version in UD v2.12;
3. We developed NLP models for the four tasks, establishing a baseline for future research aiming to enhance these models.

The rest of this paper is organized as follows: Section 2 discusses the previous work that is relevant to our work; Section 3 presents our proposed annotation guidelines for tokenization, POS tagging, morphological feature tagging, and dependency parsing dataset; Section 4 discusses the development of the Javanese dataset; Section 5 describes the experiment results using the new dataset and discussion, and finally, Section 6 presents the conclusions of our work and future work.

2. Related Works

This section discusses related works that helped us develop the Javanese treebank.

2.1. Universal Dependencies

UD is a framework for representing the grammatical structure of natural language sentences. It provides a standard set of syntactic categories and relations that can be used to analyze and compare the syntax of different languages [17]. The latest version of the UD dataset has 243 treebanks from 138 languages.

The UD dataset uses the CoNLL-U format. The CoNLL-U format is named after the conference on Computational Natural Language Learning (CoNLL), where it was first introduced. Each sentence in the CoNLL-U format is represented as a series of lines, with each line representing a single word or token in the sentence. Each line consists of 10 tab-separated fields, which provide information about the word or token, lemma, POS tag, morphological feature, and its syntactic relationships with other words or tokens in the sentence.

The UD annotation guidelines define the tag set for three tasks: 1) POS tagging, 2) morphological feature tagging, and 3) dependency parsing. For POS tagging, UD defines a tag set of 17 labels called Universal Part-of-Speech (UPOS) tags. **Appendix A** shows the UD v2 UPOS tagset. For morphological feature tagging, UD defines 24 universal features, shown in **Appendix B**. Each morphological feature has one or more values. For dependency parsing, UD defines 37 universal dependency relations between a word and its parent (see **Appendix C** for the complete list). Some language-specific dependency relations have been proposed to comply with the structure of specific languages. The newest version of the UD annotation guidelines is the UD v2.

2.2. UD Treebanks for Austronesian Languages

Javanese is an Austronesian language. In the UD dataset, three Austronesian languages are already represented: Indonesian, Tagalog, and Cebuano. The last two are from the Philippines. Regarding dataset size, only the Indonesian treebank has a decent size, which is around 169 thousand words in total. In comparison, treebanks from two other languages only have approximately 1,000 words each.

There are three Indonesian treebanks in the UD dataset. The biggest one is the UD_Indonesian-GSD [20], with 122,000 words, followed by the UD_Indonesian-CSUI [21], which consists of around 28,200 words. The smallest one is the UD_Indonesian-PUD [22–24], with approximately 19,400 words.

For Indonesian UD, some works have proposed annotation guidelines for tokenization or word segmentation, POS tagging, morphological feature tagging, and dependency parsing [22, 23]. Since Javanese and Indonesian have similar roots, we adapted the Indonesian annotation guidelines for the Javanese treebank.

2.3. Previous Work on Annotating POS Tagging for Javanese

A Javanese POS tag set of 19 tags was proposed [10]. This tag set consists of one tag for symbols, eight tags for

various punctuation, and ten tags for others. **Table 1** shows our analysis of the mapping between the UD v2 UPOS tag set and the proposed Javanese POS tag set[10]. Among 17 UD POS tags, only nine are similar to Pramudita et al.'s tags, while eight are incompatible. The differences with the other eight tags are as follows:

1. UD only has one tag, PUNCT, for all punctuation, while eight tags were defined for various punctuation[10];
2. Words like *rada* "somewhat/kind of" or *banget* "very" in UD are annotated as adverbs (ADV), while a special tag Kh was used to label them[10];
3. We also observed from the examples[12], who adopted the tagset[10], that words like *wis* "have/had" that are usually annotated as auxiliary (AUX) in UD were annotated as ADV;
4. Five UD tags are not represented or discussed[10]: 1) proper noun (PROPN), 2) numeric (NUM), 3) foreign word (X), 4) determiner (DET), and 5) particle (PART).

Table 1. The UD v2 UPOS tag set vs. Pramuditha, et al.'s tag set.

#	UD v2	Pramudita, et al.'s Tag	#	UD v2	Pramudita, et al.'s Tag
1	ADJ	Adj	10	PART	-
2	ADP	Prp	11	PRON	Pr
3	ADV	Adv, Kh	12	PROPN	-
4	AUX	Adv[12]	13	PUNCT	eight tags for various punctuation
5	CCONJ	Knj	14	SCONJ	So
6	DET	-	15	SYM	Sym
7	INTJ	Em	16	VERB	V
8	NOUN	N	17	X	-
9	NUM	-			

3. Proposed Annotation Guidelines

This section presents the proposed annotation guidelines for annotating Javanese sentences that conform to UD v2.

3.1. Language Levels

There are different speech levels in Javanese used in daily conversation. The level is generally determined based on the social status or intimacy with whom a person talks. The language level in Javanese is mainly divided into Ngoko and Krama. Ngoko is the informal/casual form, and Krama is the formal/polite form. The middle level is a continuum between Ngoko and Krama called Madya or Krama Madya. Following are more details about each language level based on those mentioned in these studies[25–27]. Theoretically, each of the following language levels also has several sublevels, but practically, most people only address these primary language levels.

1. **Ngoko**. Ngoko comes from the word Koko which refers to Ko in the word Kowe (which means "you" in casual conversation). Thus, Ngoko is the speech level used between those who are already familiar with each other or have the same social status (e.g., between classmates or siblings);
2. **Krama**. Krama comes from Sanskrit, which means "in order/properly ordered speech." People usually use this speech level to talk with their ancestors or to those with a higher status (e.g., their boss or supervisor);
3. **Madya** or **Krama Madya**. Madya, also from Sanskrit, means "middle." This level is similar to Krama but is usually used to talk to those with high status who feel at ease or strangers in the same position.

Following are the examples of applying each language level for "He or She said that his or her parents could not come here"[28].

- Ngoko: *Dheweke kandha yen wong tuwane ora bisa teka mrene.*
- Madya: *Piyambake criyos yen tiyang sepuhe mboten saged tindak mriki.*
- Krama: *Piyambakipun ngendika bilih tiyang sepuhipun mboten saged tindak mriki.*

From the above examples, each language level generally uses different vocabulary. The number of words in the

Ngoko vocabulary is larger than in the Krama vocabulary. The Madya vocabulary can be a mix of Ngoko and Krama words, Krama words, or a shortened version of Krama [27]. For example, *teng* is the shortened version of *dhateng* "to (somewhere)."

In Krama vocabulary, there are special honorific words known as Krama Inggil and Krama Andhap. Krama Inggil words are used to express respect towards the person being addressed. Unlike some words in Ngoko and Krama languages that share the same lemma but have different prefixes or suffixes, Krama Inggil's words are distinct and unique. Similarly, Krama Andhap's words are used to show humility, either towards the person being addressed or towards oneself. These words, with their embedded respect and humility, are a testament to the Javanese culture's emphasis on mutual respect and social hierarchy.

Table 2 gives some examples of Ngoko and Krama words. The first two Ngoko words (*abang* "red" and *lara* "sick") have their Krama forms, while the following two Ngoko words (*apel* "apple" and *kewan* "animal") have no Krama forms. In the Krama language, when the Krama form of a word could not be found, then the Ngoko version would be used for that word. In the next two examples, both Ngoko and Krama words use the same lemma but use different prefixes or suffixes. For *bukune* "the book," is identified as a Ngoko word since it uses the suffix -ne, while *bukunipun* as Krama uses the suffix -nipun. For *ditabrak* "be hit," it is identified as a Ngoko word since it uses the prefix di-, while *dipuntabrak* as Krama uses the prefix dipun-. The last two words (*weteng* "stomach" and *ngundang* "invite") show the example of words with the same Ngoko and Krama forms but different Krama Inggil or Krama Andhap forms.

Table 2. Some examples of Ngoko words and their associated Krama forms, along with the English translations.

English	Ngoko	Krama	Honorific
Red	*Abang*	*Abrit*	
Sick/ill	*Lara*	*Gerah*	
Apple	*Apel*	-	
Animal	*Kewan*	-	
The Book	*Bukune*	*Bukunipun*	
Be Hit	*Ditabrak*	*Dipuntabrak*	
Stomach	*Weteng*	*Weteng*	Krama Inggil: *padharan*
To Invite	*Ngundang*	*Ngundang*	Krama Inggil: *nimbali*, Krama Andhap: *ngaturi*

We annotate the Javanese language level information for each word by using the Polite feature to be tagged in the 6th column in the CoNLL-U format with the following values:

- Polite=Infm, for Ngoko words (informal)
- Polite=Form, for Krama words (formal)
- Polite=Elev, for Krama Inggil words (honorific)
- Polite=Humb, for Krama Andhap words (honorific to oneself)

Note that we do not define a level for the Madya language since we consider it the default language level. For the rest of this article, we use the abbreviation "Ng." to identify Ngoko words, and the abbreviation "Kr." for Krama words.

3.2. Tokenization and Word Segmentation

For the tokenization task, tokens are delimited by whitespace characters. However, multiword tokens (MWT) and punctuation are given special treatment.

3.2.1. Handling Multiword Tokens

MWT is a token that consists of more than one syntactic word. For Javanese, most of the MWTs contain clitic. Javanese has both proclitic and enclitic. **Table 3** shows six proclitics and four enclitics in Javanese [28], each with its language level, POS of the clitic, and an example. Most clitics act as personal pronouns (PRON). Note that clitic -e can play two roles depending on the context: as a PRON or a DET. Since the unit of annotation in UD annotation guidelines is a word, we need to split the MWT. Therefore, all clitics in **Table 3** must be separated from their main words.

3.2.2. Handling Punctuation

For punctuation, all punctuation symbols are separated from the words, except in two cases:

- Hyphen in reduplicated words. These reduplicated

Table 3. Clitics in Javanese.

Clitic	Level	POS	Example of Word with Clitic
tak-	Ngoko	PRON	takbukak "I open" = tak- "I" + bukak "open"
dak-	Ngoko	PRON	dakopenane "I take care" = dak- "I" + openane "take care"
kok-	Ngoko	PRON	kokjupuk "you take" = kok- "you" + jupuk "take"
mbok-	Ngoko	PRON	mbokpangan "you eat" = mbok- "you" + pangan "eat"
ma-	Ngoko	ADP	mangulon "to the west" = ma- "to" + kulon "west"
ke-	Ngoko	ADP	mengetan "to the east" = me- "to" + wetan "east"
-ku	Ngoko	PRON	bojoku "my wife" = bojo "wife" + -ku "my"
-mu	Ngoko	PRON	omahmu "your house" = omah "house" + -mu "your"
-ipun	Krama	PRON	ramanipun "his father" = rama "father" + -ipun "his"
-e	Ngoko	PRON	putrane "his/her son" = putra "son" + -e "his/her"
		DET	leluhure "the ancestor" = leluhur "ancestor" + -e "the"

words are not split and remain one token.

- In abbreviation. All abbreviations such as Dr., Tn. "Mr.", Ny. "Mrs." is not split and remains one token.

In Javanese, reduplication is used not only to indicate plural nouns but also for many reasons for other classes of words. **Table 4** shows examples of reduplicated words for ADJ, ADV, DET, NOUN, and VERB in Javanese. Similar characteristics are also observed for Indonesians [22].

3.3. Part-of-Speech Tagging

In our work, we adopted the POS tag set defined by UD v2, which consists of 17 tags (see **Appendix A** for the complete list). Previously, we have discussed in Section 2.3 that eight of 17 UD tags are not compatible with the proposed tag set [10]: 1) PUNCT, 2) ADV, 3) AUX, 4) PROPN, 5) NUM, 6) X, 7) DET, and 8) PART. Adapting PUNCT, PROPN, NUM, and X is relatively straightforward among those eight. PUNCT is used to label all kinds of punctuation, PROPN is for named entities, NUM is for cardinal and ordinal numbers, and X is for non-Javanese words. However, adjusting AUX, ADV, DET, and PART to Javanese words needs more discussion. Furthermore, since PRON has a strong association with DET, we will first discuss how to apply this tag for Javanese.

3.3.1. Pronoun

PRON are words that substitute for nouns or noun phrases. The UD annotation guidelines state several groups of words to be labeled as PRON. The first group is personal PRON. In Javanese, personal PRON exists for the first, second, and third person. However, not all have a lexical form for the plural PRON [28]. **Table 5** shows some examples of Javanese personal PRON for Ngoko, Krama, and Krama Inggil languages. As can be seen, for the plural PRON, only the first-person plural pronoun has a specific word that represents "we". The other plural PRON usually uses noun phrases like *kowe kabeh* which means "all of you" or *dheweke kabeh* which means "all of them".

Other word groups that are labeled as PRON in the UD annotation guidelines and suitable for Javanese words are:

- Interrogative PRON, e.g.: *apa* "what" as in *"Apa tegese?"* "What does it mean?".
- Relative PRON, e.g.: *kang* "that/which" as in *"papan kang endah"* "a place that is beautiful".
- Demonstrative PRON, e.g.: *kuwi* "that" as in *"babagan kuwi"* "about that".
- Total PRON, e.g.: *kabeh* "all" as in *"Kabeh gek kaya ngono?"* "All are like this".

3.3.2. Determiner

DET are words that modify nouns or noun phrases. Several word groups are labeled as DET:

- Article. Although in Javanese grammar there is no definition of articles, we found several Javanese words with similar roles, such as:
  - *sawijining*, as the equivalent of "a" in English.
  - -e, -ne, -ipun, -nipun, and para, as the equivalents of "the" in English.
- Demonstrative DET, e.g.: *kuwi* as in *"bocah kuwi"* or "that kid".
- Quantity DET (quantifiers), e.g.: *saperangan* "some", *akeh* "many", *kabeh* "all".

Note that depending on the context, some words can

Table 4. Reduplications in Javanese.

POS	Examples
ADJ	*sregep-sregep* "diligent", *cilik-cilik* "small"
ADV	*saapik-apike* "as well as possible", *adep-adepan* "face to face"
DET	*pinten-pinten* "several"
NOUN	*bangsa-bangsa* "nations"
VERB	*mlaku-mlaku* "travel", *nulis-nulisi* "write", *disiya-siya* "be wasted"

Table 5. Personal Pronouns in Javanese.

Type	English	Ngoko	Krama	Krama Inggil
1st-sing	I	*aku, awakku*	*kula*	*kawula, dalem*
1st-plur	we	*kita, awake dhewe*	*kula sedaya*	*kawula sedanten*
2nd-sing	you	*kowe*	*sampeyan*	*panjenengan*
2nd-plur	all of you	*kowe kabeh*	*sampeyan sedaya*	*pajenengan sedanten*
3rd-sing	he/she	*dheweke*	*piyambakipun*	*panjenenganipun*
3rd-plur	they/all of them	*dheweke kabeh*	*piyambakipun sedaya*	*panjenenganipun sedanten*

play roles both as demonstrative PRON and demonstrative DET. This is similar to total PRON, which can play roles as quantity DET.

3.3.3. Auxiliary

The AUX is not defined in Javanese reference grammar[25, 28]. We adjusted Javanese words into AUX if they fit the criteria determined by UD annotation guidelines.

- Copulas, e.g.: *yaiku* (Ng.) "be" or *inggih punika* (Kr.) "be".
- Tense-related AX. Javanese grammar has no tenses, but we can adjust words with the same meaning with examples given in UD guidelines for certain tenses. For example:
  - *bakal* (Ng.) "will", *bade* (Kr.) "will/would" for the future tense.
  - *lagi* (Ng.), *saweg* (Kr.) "be" for the present tense.
  - *wis* (Ng.), *sampun* (Kr.) "have/has/had" for the simple/past perfect tense.
- Modal-related AX. Javanese grammar also does not define modals like "can, must, may" in English. For this case, we also treat words with the same meaning in Javanese as modal. For example:
  - *kudu* (Ng.) and *mesti* (Kr.) as the equivalents of modal "must".
  - *sekudune* (Ng.), *semestine* (Kr.) as the equivalents of modal 'shall/should'.
  - *bisa* (Ng.), *saged* (Kr.) as the equivalents of modal "can/could".

3.3.4. Adverb

ADV are words that usually modify a verb, adjective, or other ADV. UD has some groups of ADV that have equivalents in Javanese:

- verb/adjective modifier, e.g: *banget* "very"
- ADJ + ly, e.g: *kanthi bungah* (ADP + ADJ) "happily"
- interrogative/relative ADV, e.g: *kok* "why", *sepira* "how"
- demonstrative ADV, e.g: *mriki* "here", *sesuk* "tomorrow", *saiki* "now"
- totality ADV, e.g.: *tansah* "always"

3.3.5. Particle

PART are function words that must be associated with another word. For Javanese, we propose to label the following words as PART:

- Negation PART, such as *ora* "not" or *boten* "not".
- Words used to emphasize something, such as *ta*, *ya*.

3.4. Morphological Feature Tagging

We propose using 13 of 24 UD v2 morphological features (see **Appendix B** for the complete list). For each feature, we consider its relevance for Javanese, and we select suitable feature values. In total, there are 31 feature-value tags that we consider relevant to Javanese grammar. **Table 6** shows the selected feature-value tags.

Among the 13 features, three are universal and not spe-

cific to Javanese grammar: Abbr, Foreign, and Typo. Feature Abbr is used for abbreviation words, feature Foreign is used for X, and feature Typo is used for misspelled words. We will discuss the other ten features in the following paragraphs.

Feature Definite distinguishes whether we are discussing something known and concrete or general/unknown. We propose using only two of five possible values for this feature: Def for definite PRON or DET and Ind for indefinite PRON or DET.

Feature Mood is applied to verbs and has 14 possible values. For Javanese, we propose using only three of them:

- Imp for imperative verbs such as for *kunceni* "lock" in *Kunceni lawange*! "Lock the door!"
- Ind for indicative mood. This mood can be considered as the default mood.
- Irr for irrealis verbs, such as for *wenehana* "if given" in *"Wenehana dhuwit ya ora gelem"* "Even though they were given money, they didn't want to accept it."[28]. The irrealis mood of the Javanese also had been discussed[29].

Feature Number is applied to DET, NOUN, and PRON. Among 11 possible values, we only use two values for Javanese: Sing for singular noun, PRON or DET and Plur for plural noun/pronoun/DET.

Feature NumType (Numeral Type) is applied to only the NUM tag. Among seven possible values, we chose only two for Javanese: Card for cardinal numbers and Ord for ordinal numbers.

Feature Person is used for personal PRON. Among five possible values defined by UD, only three are relevant: 1, 2, and 3, as already explained in Section 3.3.1 about personal PRON in Javanese.

Feature Polarity is used to mark a negation word with the value Neg. For Javanese, we use this feature for words like *ora* "not" or *durung* "not yet".

Feature Polite is used to express politeness or respect. As discussed in Section 3.1, Javanese has several language levels to express politeness: Ngoko for informal language, Krama for formal language, and Krama Inggil and Krama Andhap for honorific languages. Therefore, all possible values for Polite are relevant for Javanese, making this feature very important for Javanese.

Feature PronType has 11 possible values. For Javanese, we found eight relevant values, as shown in **Table 6.** This feature is applied to DET, PRON, or ADV. The discussion about several types of PRON and DET have been discussed in Section 3.3.1 and Section 3.3.2.

Feature Reflex describes whether the word is reflexive, i.e., refers to the subject of its clause. This feature only has one possible value: Yes. We suggest that this feature is relevant for Javanese since there is the word *dhekne* "self" is used to refer to the subject.

Feature Voice is applied to verbs. There are ten possible values for this feature, but only two values are relevant for Javanese: Act for active verbs and Pass for passive verbs.

3.5. Dependency Annotation

While annotating the dataset, we determined the dependency relations suitable for Javanese iteratively. All 37 universal dependency relations defined by UD are suitable for Javanese grammar. However, some cases need special dependency relations. Moreover, **Table 7** shows 14 subtypes or language-specific dependency relations that we propose for annotating Javanese sentences.

Among 14 proposed language-specific dependency relations (subtypes) in Table 7, 11 subtypes are applied to many languages: 1) acl:relcl, 2) csubj: outer, 3) csubj: pass, 4) flat: foreign, 5) flat: name, 6) nmod: poss, 7) nmod:tmod, 8) nsubj: outer, 9) nsubj: pass, 10) obl: agent, and 11) obl:tmod. Therefore, we can consider these 11 subtypes quite universal. For the other three subtypes: 1) advmod:emph, 2) case: adv, and 3) nmod:lmod, these subtypes are already defined for Indonesian treebanks in UD[23], and we decided to adopt them for Javanese.

Subtype *advmod:emph* is used when a word (usually a PART) is used to emphasize another word, such as for the word *ta* in *"Hla kok tetep mangkat ta, Ndhuk?"* "Why are you still going, *Ndhuk*?" that emphasizes the word *mangkat* "go". The word ta has no specific meaning in this sentence and has no corresponding translation in English.

Subtype case: adv, as it has been used for Indonesian, is used to construct adverbial phrases such as for word kanti "with" in *kanti apik* "beautifully" that together with an adjective *apik* "beautiful" produced an ADV. This ADP (preposition) + ADJ = ADV structure is similar to adding the suffix -ly to ADJ in English to construct an ADV. Subtype nmod: lmod is used for locative words in Javanese. We found that Javanese also has a similar construction to Indonesian for

Table 6. Proposed morphological features for Javanese.

#	Feature	Value	Description
1	Abbr	Yes	Abbreviation
2	Definite	Def	For definite pronouns or determiner
3	Definite	Ind	For indefinite pronouns or determiner
4	Foreign	Yes	Foreign word
5	Mood	Imp	Imperative mood
6	Mood	Ind	Indicative mood
7	Mood	Irr	Irrealis mood
8	Number	Plur	For plural nouns or pronoun
9	Number	Sing	For singular nouns or pronoun
10	NumType	Card	For cardinal number
11	NumType	Ord	For ordinal number
12	Person	1	First-person
13	Person	2	Second person
14	Person	3	Third person
15	Polarity	Neg	For negation or negative response
16	Polite	Elev	For Krama Inggil word (honorific)
17	Polite	Form	For Krama word (formal)
18	Polite	Humb	For Krama Andhap word (honorific)
19	Polite	Infm	For Ngoko word (informal)
20	PronType	Art	article
21	PronType	Dem	demonstrative pronoun/determiner/adverb
22	PronType	Emp	emphasis determiner
23	PronType	Ind	indefinite pronoun/determiner/adverb
24	PronType	Int	interrogative pronoun/adverb
25	PronType	Prs	personal pronoun
26	PronType	Rel	relative pronoun/adverb
27	PronType	Tot	total pronoun/determiner/adverb
28	Reflex	Yes	reflexive pronoun
29	Typo	Yes	for typo
30	Voice	Act	active verb
31	Voice	Pass	passive verb

some prepositions that consist of two words instead of one word for its equivalent in English. For example, ing s*ajroning* is equivalent to "in" in English. In Javanese, we only consider ing as an ADP and annotate *sajroning* as NOUN with subtype *nmod:lmod.*

4. Development of The Javanese Dataset

In this section, we explain how we built the dataset.

4.1. Selecting Sentences

As we wanted to annotate the formal Javanese text, we collected sentences from Wikipedia, grammar books, and online newspapers. So naturally, we exclude sentences from social media like Facebook or Twitter.

Initially, we only took sentences from OPUS [30], especially from WikiMatrix v1 corpus, which contains text from Wikipedia for Javanese. However, we found that some sentences had more Indonesian than Javanese words. Therefore, we fixed these sentences so that most of their words were Javanese, but this process was time-consuming. Another problem is our annotators felt that the sentences from Wikipedia were unnatural for native Javanese speakers. Therefore, we only used 150 sentences from OPUS for these two reasons and looked for other sources for the additional sentences.

To have valid Javanese sentences, we selected sentences from two Javanese grammar books [26, 28]. The sentences are perfect but mostly short. From the main grammar book for Javanese [28], we choose 100 sample sentences for specific grammatical rules relevant to our dataset. Mean-

Table 7. Proposed language-specific dependency relations (subtypes) for Javanese.

#	Subtype	Description
1	acl:relcl	for relative clause
2	**advmod: emph**	for particles that are used to emphasize a certain word
3	**case: adv**	for prepositions that become dependent on an adjective
4	csubj: outer	outer clausal subjects of predicates that are clauses
5	csubj: pass	subject clause of passive
6	flat: foreign	for named entities
7	flat: name	for named entities
8	nmod: poss	for phrase of ownership
9	**nmod: lmod**	for location words
10	nmod: tmod	for nmod that plays the role of a temporal adverbial
11	nsubj: outer	outer nominal subjects of predicates that are clauses
12	nsubj: pass	the subject of a passive sentence
13	obl: agent	for the agent of a passive clause
14	obl: tmod	for obl that plays a role as an adverbial

while, the Ngoko language was mainly discussed[26], we only selected 25 sentences since the unique words are limited.

Finally, we took 725 sentences from Solopos, online news with a section for Javanese. They have fiction and Javanese non-fiction articles. Our annotators found that sentences from this news use pure Javanese vocabulary, not mixed with Indonesian vocabulary, so they sound natural to native Javanese speakers.

4.2. Annotators and Annotation Tasks

The annotation involved six annotators: five native Javanese speakers and one non-native. Of six annotators, five with a master's degree, and one with a bachelor's degree. We have four annotation tasks for each sentence:

1) Validating and correcting the tokenization of a sentence into tokens, MWT, and words.
2) Validating and correcting the POS tag for each word in a sentence.
3) Validating and correcting the morphological feature tags for each word, including the language level tag (Ngoko or Krama language).
4) Validating and correcting the dependency parsing related data: determine the head of each word and the dependency relation between the word and its head.

Those four tasks are grouped into dependency and non-dependency-related tasks. The no-dependency-related tasks are the first three tasks mentioned above, while the dependency-related task group only consists of the last task.

4.3. Annotation Stages

We developed the dataset in two stages. In the first stage, we utilized Aksara[31], an Indonesian NLP tool that conforms to the UD, to produce the initial dataset in the CoNLLU format. Since the tokenization rules for Javanese and Indonesian are similar and use whitespace as the delimiter of tokens, most sentences are tokenized correctly. Most tokenization errors are related to MWT in Javanese and need to be fixed manually by annotators. However, the initial annotation by Aksara for POS tagging, morphological feature tagging, and dependency-related annotation was poor and needed huge corrections manually. Nevertheless, we produced the annotation for 125 sentences in this first stage. For the second stage, we engaged in an iterative process of building the dataset. We incrementally built the model using UDPipe[19], every time we completed a new 100-sentence annotation. The resulting model automatically annotates new sentences, after which corrections are made manually. In the second stage, we produced an additional 875 annotated sentences, making the final dataset size 1,000 sentences. **Figure 1** shows a dependency tree of a Javanese sentence, drawing using the CoNLL-U Viewer. We can see the tokenization, POS tagging, and dependency parsing annotation for this sentence.

4.4. Validating the Dataset

Since we have limited resources, and most annotators cannot annotate dependency-parsing-related data which was

considered more complex, one sentence was only annotated once for each task. Moreover, we arranged the annotation process so that two annotators annotated one sentence. The first annotator was responsible for non-dependency-related data; another was responsible for dependency-related data. With this scheme, we hope there is a cross-check for tokenization and POS tagging annotation since dependency annotators utilized the result of those two tasks for their work. Furthermore, to maintain the quality of the dataset, we validated the dataset using the following approaches:

- We utilized a tool provided by UD. This program will notify us if there are things that violate the UD annotation guidelines. If our dataset passes this validation tool, our dataset will be considered valid and can be uploaded to the UD repository.
- We also utilize another tool named Udapi [32], which can tell you whether a verb has more than two subjects, and so on.
- We created programs that provide statistics on the annotations for each task. For example, the program provides all possible labels for each unique word in the dataset for the POS tagging task. Using this report, we analyzed words with more than one possible POS label and decided whether the given labels were correct.
- We also conducted weekly meetings so annotators could decide on ambiguous cases found in the previous step. We also use this meeting to review our annotation guidelines for new cases that arose during the annotation process.
- Finally, we built and evaluated NLP models using the resulting dataset and UDPipe, as presented in Section 5.

4.5. Statistics of the Dataset

Table 8 shows the statistics related to the resulting dataset. We annotated 1,000 sentences, which consist of 13,723 tokens. Of those tokens, 597 are MWT. After splitting the MWT, we have a total of 14,323 words. With an average sentence length of 14.32 words/sentence, the sentences in our dataset generally consist of simple sentences. Regarding the Javanese language level of words in the dataset, of 10,375 words that are not punctuation, symbols, PROPN, and X, 4,911 words (47%) are identified with the Polite feature that represents the language level for Javanese, such as Ngoko, Krama, Krama Inggil, and Krama Andhap. **Figure 2** shows the distribution of those levels. We can see that most labeled words with language levels are Ngoko, and only a tiny portion are Krama words, with very small occurrences of Krama Inggil. Krama Andap is represented in our dataset, but since the occurrence is very small, it cannot be shown on that pie chart.

Table 8. The statistics of the dataset.

Description	Statistic
Sentence count	1,000
Token count	13,723
MWT count	597
Unique MWT count	346
Word or form count	14,323
Unique word or form count	3,789
Average sentence length (in words)	14.32
UPOS tag count	17
Morphological feature count	13
Morpholofical feature-value tag count	31
Universal dependency relation count	32
Language-specific dependency relation count	14
Total dependency relation count	46

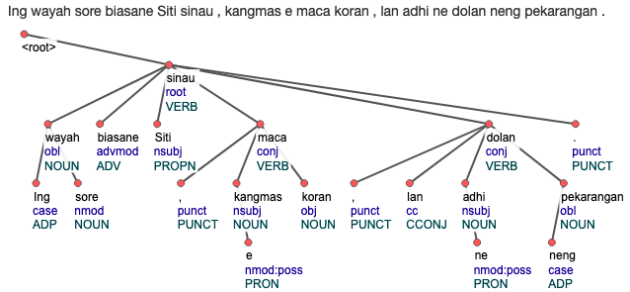

Figure 1. A dependency tree of an annotated sentence "Ing wayah sore biasane Siti sinau, kangmase maca koran, lan adhine dolan neng pekarangan" (In the afternoon, Siti usually studies, her brother reads the newspaper, and her younger sibling plays in the yard).

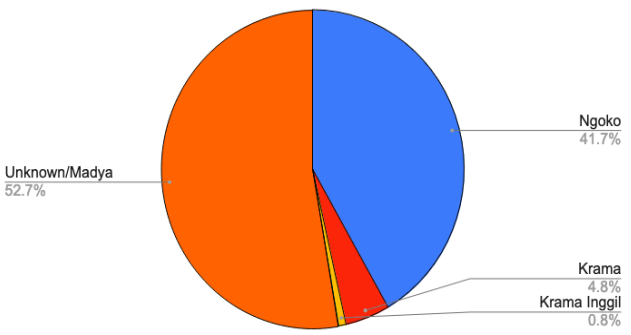

Figure 2. Distribution of words according to the Javanese language level.

The distribution also shows that most words are still unlabeled. Among unlabeled words are digit and Madya words. However, most of them are unlabeled since our annotators are unsure about their language levels. This issue will be our future work to determine the language levels of the remaining words. For POS tagging, our dataset covers all 17 tags defined by UD**. Figure 3** shows our dataset's distribution of all UD POS tags. The most frequent POS tags are NOUN, PUNCT, VERB, PROPN, and PRON, while the least frequent tags are SYM (symbol) and INTJ (interjection).

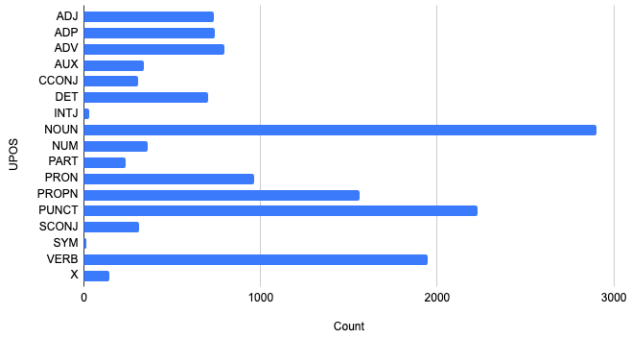

Figure 3. Distribution of Universal POS tags.

For morphological feature tags, of 13 morphological features proposed in Section 3.4, our dataset has examples for all of them, and among 31 proposed morphological feature-value tags, all feature-value tags are also represented. **Table 9** shows the distribution of feature-value tags in the dataset, along with the top three words for each feature-value tag. The two most frequent feature-value tags are Polite=Infm for informal words (Ngoko) with the number occurrences of 4,326, and Number=Sing which represents a singular noun that occurs 3,305 times.

For dependency relations, of 37 universal dependency relations defined by UD, 32 are represented in the dataset, with the distribution shown in **Table 10**. The universal dependency relations that are not represented: are *dislocated*, *expl* (expletive), *list*, *orphan* and *reparandum*. Of the 14 subtypes we proposed in Section 3.5, all are represented in the dataset. However, seven dependency relations occur less than ten times: *clf*, *dep*, *goes with*, *iobj*, *csubj:outer*, *csubj:pass*, and *nsubj:outer*.

5. Results and Discussion

To evaluate the quality of the resulting dataset, we conducted experiments to build models for four NLP tasks: 1) tokenization, 2) POS tagging, 3) morphological feature tagging, and 4) dependency parsing.

5.1. Building Model with UDPipe

We used UDPipe v1.0 to build the model[19]. UDPipe is a language-agnostic toolkit for 1) sentence segmentation, 2) tokenization, 3) lemmatization, 4) tagging, and 5) dependency parsing of natural language texts. For tagging, it creates a model for UPOS, language-specific POS (XPOS), and morphological feature (FEAT) tagging. The recent version of UDPipe is v2.0[18]. It uses multilingual BERT as the contextualized word embedding[33].

Since our Javanese dataset does not have data for lemmatization and XPOS tagging (we left the column LEMMA and XPOS empty), we only evaluate the dataset for four tasks: 1) tokenization, 2) POS tagging, 3) morphological feature tagging, and 4) dependency parsing.

5.2. Evaluation Method

Since the size of the Javanese dataset is very small, we used the 10-fold cross-validation method. For each fold, we trained the model with a training dataset of around 12,900 words. After that, we test the model using a test dataset of approximately 1,400 words. For tokenization, UDPipe will produce the accuracy for four levels of tokenization:

1. Sentence segmentation: how accurately the model splits the text into sentences.
2. Tokenization: how accurate the model is in splitting sentences into tokens based on the whitespace and punctuation (as discussed in Section 3.2).
3. Multiword tokenization: how accurately an MWT is recognized and split into several words.
4. Word segmentation: how accurate the resulting words are.

For POS tagging and morphological feature tagging tasks, UDPipe produces the accuracy for two conditions: tagging with gold tokenization and tagging with automatic tokenization done using the tokenization model built using our dataset. For dependency parsing, UDPipe provides three scenarios:

- Parsing from raw text with computed tokenization and computed POS tags
- Parsing from gold tokenization with computed POS

Table 9. The distribution of the 31 UD FEAT tags in our dataset, along with the three most frequent lemmas for each POS.

#	Feature	Value	Count	The Three Most Frequent Words
1	Abbr	Yes	32	*isa -> bisa* "can", *ra -> ora* "not", *ki -> iki* "this"
2	Definite	Def	356	*e* "the", *para* "the", *ipun* "the"
3	Definite	Ind	9	*sawijining* "a", *satunggaling* "a"
4	Foreign	Yes	170	rock, eutanasia, *penerbangan* "flight"
5	Mood	Imp	7	*cekake* "check", *kunceni* "lock", *resiki* "clean"
6	Mood	Ind	1925	*ana* "exist", *dadi* "become", *gawe* "do"
7	Mood	Irr	8	*jupukna* "if being taken", *kandhanana* "if being told", *tukokna* "if been bought"
8	Number	Plur	119	*saperangan* "some", *para* "the", *akeh* "many"
9	Number	Sing	3305	*e* "the", *aku* "I", *ku* "my"
10	NumType	Card	361	*siji* "one", *rong* "two", *sak* "one"
11	NumType	Ord	14	*kapisanan* "first", *kapitu* "seventh", *katiga* "third"
12	Person	1	237	*aku* "I", *ku* "my", *dak* "I"
13	Person	2	49	*mu* "your", *kowe* "you", *awakmu* "you"
14	Person	3	202	*e* "his/her/its", *dheweke* "he/she", *ipun* "his/her/its"
15	Polarity	Neg	161	*ora* "not", *durung* "not yet", *mboten* "not"
16	Polite	Elev	83	*nalika* "when", *panjenenganipun* "he/she", *panjenengan* "you"
17	Polite	Form	493	*ingkang* "which", *punika* "that/is", *ipun* "the"
18	Polite	Humb	9	*nyuwun* "request", *nggih* "yes", *kulanuwun* "excuse me"
19	Polite	Infm	4326	*e* "the", *ing* "at", *lan* "and"
20	PronType	Art	365	*e* "the", *para* "the", *ipun* "the"
21	PronType	Dem	410	*kuwi* "that", *iki* "this", *iku* "that"
22	PronType	Emp	17	*dhewe* "itself", *piyambak* "itself"
23	PronType	Ind	43	*saperangan* "some", *akeh* "many", *maneka* "various"
24	PronType	Int	43	*kok* "why", *apa* "what", *kena apa* "why"
25	PronType	Prs	491	*e* "he/she/it", *aku* "I", *ku* "my"
26	PronType	Rel	357	*kang* "that", *sing* "that", *ingkang* "which/that"
27	PronType	Tot	58	*kabeh* "all", *saben* "every", *tansah* "always"
28	Reflex	Yes	3	*diri* "self", *dhekne* "self"
29	Typo	Yes	3	*taunn -> taun* "year", *kula warga –> kulawarga* "family"
30	Voice	Act	1536	*ana* "exist", *dadi* "become", *gawe* "do"
31	Voice	Pass	406	*katon* "be seen", *kelingan* "be remembered", *diripta* "be created"

tags

- Parsing from gold tokenization with gold POS tags

We will only report the result using gold tokenization for both POS tagging and dependency parsing.

5.3. Evaluation Metrics

We used the evaluation metrics provided by UDPipe. It produces the Precision, Recall, and F1-score for the tokenization task, while for tagging (both POS and morphological feature), it gives only the F1-score. These measures are based on the number of True Positive (TP), False Positive (FP), and False Negative (FN) results for each result.

$$Precision = \frac{|TP|}{|TP| + |FP|} \quad (1)$$

$$Recall = \frac{|TP|}{|TP| + |FN|} \quad (2)$$

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (3)$$

Finally, attachment scores are produced to evaluate dependency parsing. The attachment score is the percentage of words with correct heads or labels. There are two kinds of attachment scores: Unlabeled Attachment Score (UAS) and Labeled Attachment Score (LAS).

$$UAS = \frac{number\ of\ tokens\ wit\ correct\ eads}{number\ of\ tokens} \quad (4)$$

$$LAS = \frac{number\ of\ tokens\ wit\ correct\ eads\ and\ labels}{number\ of\ tokens} \quad (5)$$

Table 10. The distribution of 32 UD universal dependency relations (deprel) in our dataset, along with the distribution of 14 language-specific dependency relations (subtype).

#	Deprel	Count	#	Deprel	Count	#	Subtype	Count
1	acl	123	17	discourse	33	1	acl:relcl	256
2	advcl	629	18	fixed	17	2	advmod:emph	47
3	advmood	1071	19	flat	83	3	case:adv	10
4	amod	245	20	goeswith	2	4	csubj:outer	1
5	appos	102	21	iobj	5	5	csubj:pass	2
6	aux	300	22	mark	308	6	flat:foreign	24
7	case	735	23	nmod	1177	7	flat:name	549
8	cc	309	24	nsubj	1217	8	nmod:lmod	45
9	ccomp	22	25	nummod	248	9	nmod:poss	291
10	clf	7	26	obj	494	10	nmod:tmod	15
11	compound	27	27	obl	851	11	obl:agent	30
12	conj	423	28	parataxis	81	12	obl:tmod	133
13	cop	39	29	punct	2229	13	nsubj:outer	6
14	csubj	31	30	root	1000	14	nsubj:pass	175
15	dep	1	31	vocative	48			
16	det	683	32	xcomp	197			

5.4. Discussion

Table 11 shows the experiment results for the tokenization task. We can see that the model has achieved an excellent F1-score of 99.53% in separating punctuation from the token. However, the model performance in dealing with the MWT, especially the Javanese clitics, is relatively low, with an F1-score of only 72.01%. We suspect the cause is due to the small number of MWTs in the dataset, i.e., only 597 of 13,721 tokens. Nevertheless, since the occurrence of MWTs in our dataset is small, the final accuracy for word segmentation is still outstanding, with an F1-score of 97.11%.

Table 11. Experiment results for tokenization task.

Task	Precision (%)	Recall (%)	F1-Score (%)
Tokenizer tokens	99.60	99.47	99.53
Tokenizer multiword tokens	82.36	64.33	72.01
Tokenizer words	97.64	96.59	97.11
Tokenizer sentences	95.12	96.72	95.90

Next, **Table 12** shows the result for POS tagging and morphological tagging tasks. The POS tagging model and morphological feature model with gold tokenization scenario have an F1-score of 87.22% and 86.66%, respectively. We consider these results very good since the dataset size is quite small.

Finally, the dependency parsing task's results can be seen in **Table 13**. For the first scenario, parsing from gold tokenization with computed POS tags, the UAS and LAS are very low, with UAS of only 70.49% and LAS of 60.44%. We can see that the accuracy of the POS tag significantly affects the parser's ability to parse sentences correctly. For the second scenario, parsing from gold tokenization with gold POS tags, the model achieves UAS of 77.08% and LAS of 71.21%. Meanwhile, the experiment results with similar experiments for the UD_Indonesian-PUD treebank that has around 19.400 words[23], has UAS of 82.59% and LAS of 79.83%. That experiment also used the 10-fold cross-validation and UDPipe to train and evaluate the models. Compared to the experiment with a 1.35 times bigger dataset, our results are pretty good.

The experiment results show that our dataset can be used to train Javanese NLP models with excellent tokenization accuracy and moderate accuracy for POS and morphological feature tagging. But unfortunately, it is not satisfactory for the tokenization of MWT and dependency parsing.

6. Conclusions and Future Work

In this section, we present the conclusion and future work.

6.1. Conclusions

In this work, we proposed the annotation guidelines for the Javanese dataset that conform to UD, which consists of

Table 12. Experiment results for POS and morphological feature tagging task.

Description	F1-Score (%)
POS tagging - with gold tokenization	87.22
Morphological features tagging - with gold tokenization	86.66

Table 13. Experiment results for dependency parsing task.

Description	UAS (%)	LAS (%)
Parsing from gold tokenization with computed POS tags	70.49	60.44
Parsing from gold tokenization with gold POS tags	77.08	71.21

annotation guidelines for tokenization, POS tagging, morphological feature tagging, and dependency parsing. For tokenization annotation, we highlighted the importance of handling clitics. For POS tagging annotation, we gave examples of Javanese words that belong to every tag defined in the UD POS tagset. For morphological feature tagging, we proposed using 31 feature-value tags for Javanese. Finally, for dependency parsing, we proposed 14 language-specific dependency relations.

Furthermore, we also built a dataset that complies with the proposed annotation guidelines. This dataset consists of 1,000 sentences and 14,323 words. Our dataset shows how to tokenize sentences and MWT in Javanese. The dataset also consists of words that represent all 17 POS tags defined by UD. The dataset represents all feature-value tags proposed for morphological features. As for dependency annotation, of 37 universal dependency relations defined by UD, 32 dependency relations are present in the dataset. All of the 14 language-specific dependency relations we proposed are represented in the dataset.

To evaluate the dataset quality, we built the NLP model for tokenization, POS tagging, morphological feature tagging, and dependency parsing. Since the dataset is very small, we conducted experiments using the 10-fold cross-validation method. In addition, we trained and evaluated accuracy using UDPipe[19]. The experiment results show that the NLP models built using our dataset produced a very high F1 score for the tokenization of tokens, syntactic words, and sentence tasks, a good F1 score for POS, and morphological feature tagging. Unfortunately, it has a low score for the tokenization of MWT and dependency parsing.

6.2. Future Works

The dataset we built consists of 1,000 sentences and 14,323 words, which is relatively small. We will add more sentences in the future. With a bigger dataset, the accuracy of tokenization of MWT and dependency parsing tasks using our dataset will be improved.

Furthermore, we will conduct a study on morphological analysis for Javanese and annotate the LEMMA column that is still empty in the current version of the dataset. Determining a lemma requires a more profound knowledge of the language, so we plan to collaborate with Javanese linguists to achieve this goal.

Another essential improvement for the dataset is to add the missing language level labels for approximately 50% of the words. For this work, we will consult the existing Javanese dictionary.

Moreover, we consider using the transfer learning method to build a better model for POS tagging, morphological feature tagging, and dependency parsing. Transfer learning has proven beneficial for low-resource languages like Javanese.

Author Contributions

Conceptualization, I.A.; methodology, I.A.; data curation, I.A., A.Y., D.T., and A.D.; writing—original draft preparation, I.A. and A.Y.; writing—review and editing, I.A. and D.Z.; funding acquisition, I.A. and D.Z.

Funding

This research is funded by Directorate of Research and Development, Universitas Indonesia under Hibah PUTI 2022 (Grant No. NKB-1384/UN2.RST/HKP.05.00/2022).

Institutional Review Board Statement

Not applicable.

Informed Consent Statement

Not applicable.

Data Availability Statement

The dataset is available on the Universal Dependency website.

Acknowledgments

We thank Putri Rizkiyah and Sri Hartati Wijono for their role as annotators in the initial stage of the annotation project. In addition, we also thank Prof. Ahmad Nizar Hidayanto who gave feedback for the first draft.

Conflicts of Interest

The authors declare no conflict of interest.

Appendix A

Universal Part-of-Speech (UPOS) Tagset

Table A1. The UD v2 UPOS tagset.

UPOS Tag	Word Class	UPOS Tag	Word Class
ADJ	Adjective	PART	Particle
ADP	Adposition	PRON	Pronoun
ADV	Adverb	PROPN	Proper noun
AUX	Auxiliary	PUNCT	Punctuation
CCONJ	Coordinating conjuction	SCONJ	Subordinating conjunction
DET	Determiner	SYM	Symbol
INTJ	Interjection	VERB	Verb
NOUN	Noun	X	Other
NUM	numeral		

Appendix B

Morphological Features Tagset

Table A2. List of the 24 UD v2 universal morphological features.

Abbr	Degree	Number	PronType
Animacy	Evident	NumType	Reflex
Aspect	Foreign	Person	Tense
Case	Gender	Polarity	Typo
Clusivity	Mood	Polite	VerbForm
Definite	NounClass	Poss	Voice

Appendix C

Dependency Relation Tagset

Table A3. List of the 37 UD v2 universal universal dependency relations.

acl	ccomp	discourse	mark	punct
advcl	clf	dislocated	nmod	reparandum
advmod	compound	expl	nsubj	root
amod	conj	fixed	nummod	vocative
appos	cop	flat	obj	xcomp
aux	csubj	goeswith	obl	
case	dep	iobj	orphan	
cc	det	list	parataxis	

References

[1] Eberhard, D.M., Simons, G.F., Fennig, C.D., 2022. Ethnologue: Languages of the world, 25th ed. SIL International: Dallas. pp. 1–760.

[2] Aji, A.F., Winata, G.I., Koto, F., et al., 2022. One country, 700+ Languages: NLP challenges for underrepresented languages and dialects in Indonesia. In: Muresan, S., Nakov, P., Villavicencio, A. (eds.). Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics: Dublin, Ireland. pp. 7226–7249. DOI: https://doi.org/10.18653/v1/2022.acl-long.500

[3] Indurkhya, N., Damerau, F.J., 2010. Handbook of natural language processing, 2nd ed. Chapman and Hall: New York. pp. 1–704. DOI: https://doi.org/10.1201/9781420085938

[4] Krisnawati, L.D., Mahastama, A.W., 2018. A Javanese syllabifier based on its orthographic system. Proceedings of the International Conference on Asian Language Processing (IALP) 2018; Bandung, Indonesia; 15–17 November 2018. pp. 244–249. DOI: https://doi.org/10.1109/IALP.2018.8629173

[5] Wijono, S.H., Alhamidi, M.R., Hilman, M.H., et al., 2021. Canonical segmentation using affix charac-

ters as a unit on transformer for Javanese language. Proceeding of the 2021 6th International Workshop on Big Data and Information Security (IWBIS); Depok, Indonesia; 23–25 October 2021. pp. 67–72. DOI: https://doi.org/10.1109/IWBIS53353.2021.9631839

[6] Cahyani, D.E., Utami, L.M.T, Setiadi, H., 2019. Clustering of Javanese news in krama alus level with Javanese stemming. Proceeding of the 2019 International Conference on Information and Communications Technology (ICOIACT); Yogyakarta, Indonesia; 24–25 July 2019. pp. 462–467. DOI: https://doi.org/10.1109/ICOIACT46704.2019.8938438

[7] Nq, M.A., Manik, L.P., Widiyatmoko, D., 2020. Stemming Javanese: Another adaptation of the Nazief-Adriani algorithm. Proceeding of the 2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI); 10–11 December 2020. pp. 627–631. DOI: https://doi.org/10.1109/ISRITI51436.2020.9315420

[8] Adriani, M., Asian, J., Nazief, B., et al., 2007. Stemming Indonesian: A confixed-stripping approach. ACM Transactions on Asian Language Information Processing (TALIP). 6(4), 1–33. DOI: https://doi.org/10.1145/1316457.1316459

[9] Ramadhan, F.A., Suryani, A.A., Bijaksana. M.A., 2020. Part of speech tagging in Javanese using support vector machine method. e-Proceeding of Engineering. 7(2), 1–8. Available from: https://jitl.web.id/index.php/engineering/article/view/13089

[10] Pramudita, H.R., Utami, E., Amborowati, A., 2016. Effects of rule-based part of speech tagging and distribution maximum entropy probability for Javanese krama. Jurnal Buana Informatika. 7(4), 235–244. DOI: https://doi.org/10.24002/jbi.v7i4.764

[11] Pratama, R.A., Suryani, A.A., Maharani, W., 2020. Part of speech tagging for Javanese language with hidden markov model. Journal of Computer Science and Informatics Engineering (J-Cosine). 4(1), 84–91. DOI: https://doi.org/10.29303/jcosine.v4i1.346

[12] Zilziana, A., Suryani, A.A., Asror, I., 2020. Part of speech tagging for Javanese using conditional random fields method. e-Proceeding of Engineering. 7(2), 8103–8111.

[13] Ratnaparkhi, A., 1996. A maximum entropy model for part-of-speech tagging. Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP); Philadelphia, USA; 17–18 May 1996. pp. 133–142. Available from: https://aclanthology.org/W96-0213.pdf

[14] Rabiner, L., Juang, B., 1986. An introduction to hidden markov models. IEEE ASSP Magazine. 3(1), 4–16. DOI: https://doi.org/10.1109/MASSP.1986.1165342

[15] Hearst, M.A., Dumais, S.T., Osuna, E., et al., 1998. Support vector machines. IEEE Intelligent Systems and Their Applications. 13(4), 18–28. DOI: https://doi.org/10.1109/5254.708428

[16] Sutton, C., McCallum, A., 2012. An introduction to conditional random fields. Foundations and Trends® in Machine Learning. 4(4), 267–373. DOI: https://doi.org/10.1561/2200000013

[17] Nivre, J., de Marneffe, M.C., Ginter, F., et al., 2020. Universal Dependencies v2: An Evergrowing Multilingual Treebank Collection. In Language Resources and Evaluation (LREC). pp. 4034–4043.

[18] Straka, M., 2018. UDPIPE 2.0 prototype at Conll 2018 UD shared task. Proceeding of the CoNLL 2018 - SIGNLL Conference on Computational Natural Language Learning. Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies; Brussels, Belgium; 31 October–1 November 2018. pp. 197–207. DOI: https://doi.org/10.18653/v1/K18-2020

[19] Straka, M., Hajič, J., Straková, J., 2016. UDPipe: Trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16); Portorož, Slovenia; 23–28 May 2016. pp. 4290–4297. Available from: https://aclanthology.org/L16-1680.pdf

[20] McDonald, R., Nivre, J., Quirmbach-brundage, Y., et al., 2013. Universal dependency annotation for multilingual parsing. Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers); Sofia, Bulgaria; 4–9 August 2013. pp. 92–97. Available from: https://aclanthology.org/P13-2017.pdf

[21] Alfina, I., Budi, I., Suhartanto, H., 2020. Tree rotations for dependency trees: Converting the head-directionality of noun phrases. Journal of Computer Science. 16(11), 1585–1597. DOI: https://doi.org/10.3844/jcssp.2020.1585.1597

[22] Alfina, I., Dinakaramani, A., Fanany, M.I., et al., 2019. A gold standard dependency treebank for Indonesian. Proceedings of the 33rd Pacific Asia Conference on Language, Information and Computation (PACLIC 33); Hakodate, Japan; 13–15 September 2019. pp. 1–9. Available from: https://www.researchgate.net/publication/334470091_A_Gold_Standard_Dependency_Treebank_for_Indonesian

[23] Alfina, I., Zeman, D., Dinakaramani, A., et al., 2020. Selecting the UD v2 Morphological Features for Indonesian Dependency Treebank. Proceedings of the 2020 International Conference of Asian Language Processing (IALP); Kuala Lumpur, Malaysia; 4–6 December 2020. pp. 104–109. DOI: https://doi.org/10.1109/IALP51396.2020.9310513

[24] Zeman, D., Hajič, J., Popel, M., et al., 2018. CoNLL 2018 shared task: Multilingual parsing from raw text to universal dependencies. In: Zeman, D.,

Hajič, J. (eds). Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. Association for Computational Linguistics; Brussels, Belgium. pp. 1–21. DOI: https://doi.org/10.18653/v1/K18-2001

[25] Robson, S., 2014. Javanese grammar for students, a graded introduction, 3rd ed. Monash University Publishing: Clayton, Australia. pp.1–122.

[26] Suwadji. (2013). Ngoko Krama. Kementerian Pendidikan dan Kebudayaan Badan. Pengembangan dan Pembinaan Bahasa Balai Bahasa Provinsi Daerah Istimewa Yogyakarta.

[27] Wolff, J.U., Poedjosoedarmo, S., 1982. Communicative codes in central java (Volume 113–116). Southeast Asia Program, Department of Asian Studies, Cornell University: New York. pp. 1–197.

[28] Wedhawati, Nurlina, W.E.S., Setiyanto, E. (Eds.). 2006. Tata bahasa jawa mutakhir. Pusat Bahasa, Departemen Pendidikan Nasional: Jakarta. pp. 1–586.

[29] Adelaar, K.A., Himmelmann, N.(Eds.). 2004. The Austronesian languages of Asia and Madagascar. Routledge: London. pp. 1–864. DOI: https://doi.org/10.4324/9780203821121

[30] Tiedemann, J., 2012. Parallel data, tools and interfaces in OPUS. Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC 2012); Istanbul, Turkey; 21–27 May 2012. pp. 2214–2218. Available from: http://www.lrec-conf.org/proceedings/lrec2012/pdf/463_Paper.pdf

[31] Hanifmuti, M.Y., Alfina, I., 2020. Aksara: An Indonesian morphological analyzer that conforms to the UD v2 annotation guidelines. Proceedings of the 2020 International Conference of Asian Language Processing (IALP); Kuala Lumpur, Malaysia; 4–6 December 2020. pp. 86–91. DOI: https://doi.org/10.1109/IALP51396.2020.9310490

[32] Popel, M., Žabokrtský, Z., Vojtek, M., 2017. Udapi: Universal API for universal dependencies. Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017); Gothenburg, Sweden; 22 May 2017. pp. 96–101. Available from: https://aclanthology.org/W17-0412.pdf

[33] Devlin, J., Chang, M.W., Lee, K., et al., 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers); Minneapolis, Minnesota; 2–7 June 2019. pp. 4171–4186. DOI: https://doi.org/10.18653/v1/N19-1423