Towards Construction of an Error-Corrected Corpus of Indonesian Second-Language Learners' Sentences

(Corresponding author: Budi Irmawati)
Nara Institute of Science and Technology
Email: yzakodek@yahoo.com
(Yuji Matsumoto[1] and Mamoru Komachi[2])
[1]Nara Institute of Science and Technology
[2]Tokyo Metropolitan University
Email: [1]*matsu@is.naist.jp*, [2]*komachi@tmu.ac.jp*

Abstract

Most works on error-correction has been done on resource-rich language. However, even though corpus-based approach for Indonesian error correction has been studied, the availability of language resources for some languages such as Indonesian is limited. This study, using raw data taken from a multi-lingual language learning and language exchange Social Network Service (SNS), is a first step toward developing Indonesian learner corpora. We constructed the corpus by aligning learners' sentences with its corresponding native correction while morphological information for each word was assigned using automatic POS tagger. To build confusion matrix to improve the automatic alignment of the remaining data, we corrected some of the hand corrected alignment sentences. Then assigning the alignment label used the confusion matrix as a reference. Finally, the annotated pairs of errors and corrected words were analyzed and classified into error types based on the Part of Speech (POS) information. These pairs can be used as language resources.

The experiment shows that the confusion matrix extracted from the hand corrected alignment improved the alignment precision about 19.02%. This preliminary experiment used the corpus to develop a baseline for our future work on an automatic grammatical error detection system and automatic dependency annotation.

Keywords: error-corrected learner corpus, error annotation, second language learner, under resource language, corpus construction, Indonesian language learner, language exchange

1 Introduction

Recently, many Natural Language Processing (NLP) research benefit from learner corpora as resources (Ng et al., 2013). While some grammatical errors made by second language (L2) learners overlap with the errors made by native speakers, some learner errors are not covered in native speaker errors. In writing, also, second language learners make errors that occur comparatively infrequently by native speakers (Leacock et al., 2014) because the learners" first language (L1) also influences how they develop a sentence. Therefore, working on native speaker corpora is not a highly suitable basis for error detection and correction of L2 learner writings.

In English, many people worked on preparing corpora (Dahlmeier et al., 2013, Leacock et al., 2014, Marcus et al., 1993, Rozovskaya and Roth, 2010). Other learner corpora are also available such as Korean (Lee et al., 2012), Arabic (Abuhakema et al., 2008), German (Boyd, 2010), and Czech (Hana et al., 2010).However, for some under-developed languages such as Indonesian, these corpora are unavailable. Hence, for developing a language learning support system, tailoring second language learning resources of Indonesian is especially useful. This type of language resource helps language teachers in understanding the types of learner problems, designing course materials, providing feedback about mistaken grammar or word choice, constructing a confusion matrix for L2 learners, and emphasizing the use of Indonesian words instead of borrowed words from other languages.

Manual work in developing a learner corpus is laborious and error-prone, while rule-based machine annotation is too coarse and inaccurate. As an initial effort to develop an error-annotated learner corpus of Indonesian language, we combined manual and automated-based techniques. We extracted learners" writings from a language learning Social Networking Service (SNS), Lang-8[1], as raw data.

[1]http://lang-8.com

as raw data. Lang-8 is a website, where learners wrote journals and native speakers manually highlighted and corrected the errors, sentence by sentence.

After some pre-processing cleaning, we aligned words in the learners" sentences with words in the native-corrected sentences to identify error positions and the incorrect words in a two-step procedure. In the first step, we aligned the word with dynamic programming and some heuristic rules and then asked a native speaker to correct the results. Then, we extracted a confusion matrix from the aligned corrected sentences. In the second step, an automatic procedure aligned sentence pairs and assigned error tags based on the confusion matrix. We called these procedures ***Rule-based*** and *Hybrid (Rule-Based improved by the confusion matrix)* respectively. Our automatic work focused on one word-to-one-word error alignment, because we do not have a gold standard for phrase-based alignment of manually annotated data.

This alignment covered spelling errors, unnecessary words, omitted words, affixations errors, and replacement error types. The precision of the alignment increases from 70.39% in the *Rule-based* to 89.41% in the *Hybrid* procedure. We also carried out an experiment to show that semantic and syntactic analysis using a short window size is still difficult for human. We performed a preliminary experiment to identify the error types using the alignment data. We noticed that some error types show poor accuracy.

The next section briefly reviews related work on statistical alignment in monolingual corpora. Section 3 describes how we prepared and pre-processed the data. Section 4 explains two main experiments to show how the re-correction improved alignment precision and the human judgement of the error position and error type. This also describes a preliminary automated error detection system using this corpus. In Section 5, we describe the experimental results and we point out the importance of this corpus and future directions for our work in section 6.

2 Related Works

Many learner corpora, especially in English, were available as language resources. Starting from 2011, the NLP community has come together to provide a shared task in grammatical error corrections. The advantage is all the participants use the same training and test sets, and the same evaluation metrics (Leacock et al., 2014). Instead of the training data and the test data provided by the organizer, the participants can also use other resources as long as they are available publicly.

Related researches for annotated learner"s sentences have been done in some languages such as English (Fraser and Marcu, 2006, Izumi et al., 2005), Korean (Lee et al., 2012), and German (Boyd, 2010). Hana et al. (2010) work in developing a learner corpus for Czech from handwritten document that is transcribed into HTML. They converted into PML format, and then an annotator manually corrects the document.

Nagata et al. (2011) did other work on learner error corpus. They created an English learner corpus that was manually error-tagged and shallow-parsed. The error annotation is given using XML syntax by tagging a word or a phrase that contains any errors while a missing word is inserted in the missing word position.

As for Indonesian corpus, in general, some Indonesian corpora are available online such as bilingual Indonesian-English parallel corpus named Identic corpus (Larasati, 2012), the 1 Million POS Tagged corpus[2] that contains about 39 thousand sentences, and Indonesian corpora repository (Manurung et al., 2010). To the best of our knowledge, no error-corrected annotated of earner corpus for Indonesian language is currently available.

[2] `http://panl10n.net/english/OutputsIndonesia2.htm`

Learner's sentence		Saya benar-benar tertalik dari manganya selama panjang. (I was really attracted from the manga for a long time.)
Native's Correction	1	❒ Selama ini saya benar-benar tertarik dengan manga ini. ✍ **[f-blue]**Selama ini saya benar-benar tertarik dengan manga ini**[/f-blue]**.
	2	❒ Saya benar-benar tertarik ~~dari~~ dengan manga~~nya~~ ini selama ~~panjang~~ ini. ✍ Saya benar-benar tertarik **[sline]**dari**[/sline]** **[f-blue]**dengan**[/f-blue]** manga**[sline]**nya**[/sline]** **[f-red]**ini**[/f-red]** selama **[sline]**panjang**[/sline]** **[f-blue]**ini**[/f-blue]**.
	3	❒ Saya benar-benar tertalik dari manganya selama panjang. Saya benar-benar tertarik dengan manga ini. ✍ Saya benar-benar**[f-red]** tertalik**[/f-red]** **[f-red]**dari manganya selama panjang**[/f-red]**. Saya benar-benar **[f-blue]**tertarik dengan**[/f-blue]** **[f-blue]**manga ini**[/f-blue]**.

✍: Lang-8's raw data format ❒: website's view

Table 1: Example of Learner's Sentence

3 Data Preparation

We used the Lang-8 written by Indonesian learners[3] that crawled in 2011[4]. Lang-8 is a multi-lingual language learning and language exchange SNS whose entries are currently written by learners from 180 countries. In Lang-8, learners can write a free topic journal that consists of some sentences. Aside of writing a journal, learners are encouraged to correct another journal related to their first language. Therefore, almost all sentences in Lang-8 consist of learners" sentences and will have a correction from native speakers. It is possible to have more than one correction for one sentence because more than one native speaker can make corrections to one journal.

These data have extraneous content, noises such as a native"s tags, comments, and sentences in the learner"s first language. A native"s tags are the tags produced when a native speaker highlighted and corrected the learner"s sentences. Table 1 shows an example of these noises along with how the native speakers highlighted and corrected the sentence. The native"s tags are inside square brackets and the comment or L1 sentence is in the parentheses in the first row. The symbol ❒ indicates the website view of the native corrected sentence while the symbol ✍ indicates a sentence in the raw data with native"s tags. In example 1, the native speaker only highlighted the correction sentence. In example 2, the native speaker crossed out the incorrect words and highlighted the corrected words with different colors. With a different style of correction, in example 3, the native speaker rewrote and highlighted the sentence, then wrote the correction in the second line. He/she highlighted the original words and the words that they corrected with a different color.

We cleaned these noises automatically by deleting the native"s tags; then, we excluded the sentences written in other languages. Originally, the data consisted of 783 journals written by 107 learners from 15 different countries. These journals contained 6,559 learners" sentences or 77,201 words with 8,673 word types. Because some sentences had two or more corrections, after the cleaning process, this resulted in 7,420 sentence pairs.

To produce high-quality native-corrected sentences, we asked a native speaker to check the sentence pairs manually. First, a native speaker re-corrected spelling and punctuation errors; re-corrected words that violated the multi-word expression rules (a multi-word expression is written as one word if it is surrounded by both a prefix and suffix); and replaced out-of-vocabulary words of corrected sentences. In Lang-8, some native speakers wrote corrected sentences in an informal writing style. Thus, we were required to re-write them and discarded sentence pairs that could not be rewritten. These re-correction processes referred to Indonesian grammar books (Alwi, 2000, Sneddon et al., 2010).

To keep the information about the writer for later functionality, each sentence has a journal id that refers to a topic, learner id, and L1, and the corrected sentence. The result of aligned sentences has

[3]Language is initially identified via the author profile as metadata and then checked by human.

[4]`http://cl.naist.jp/nldata/lang-8`

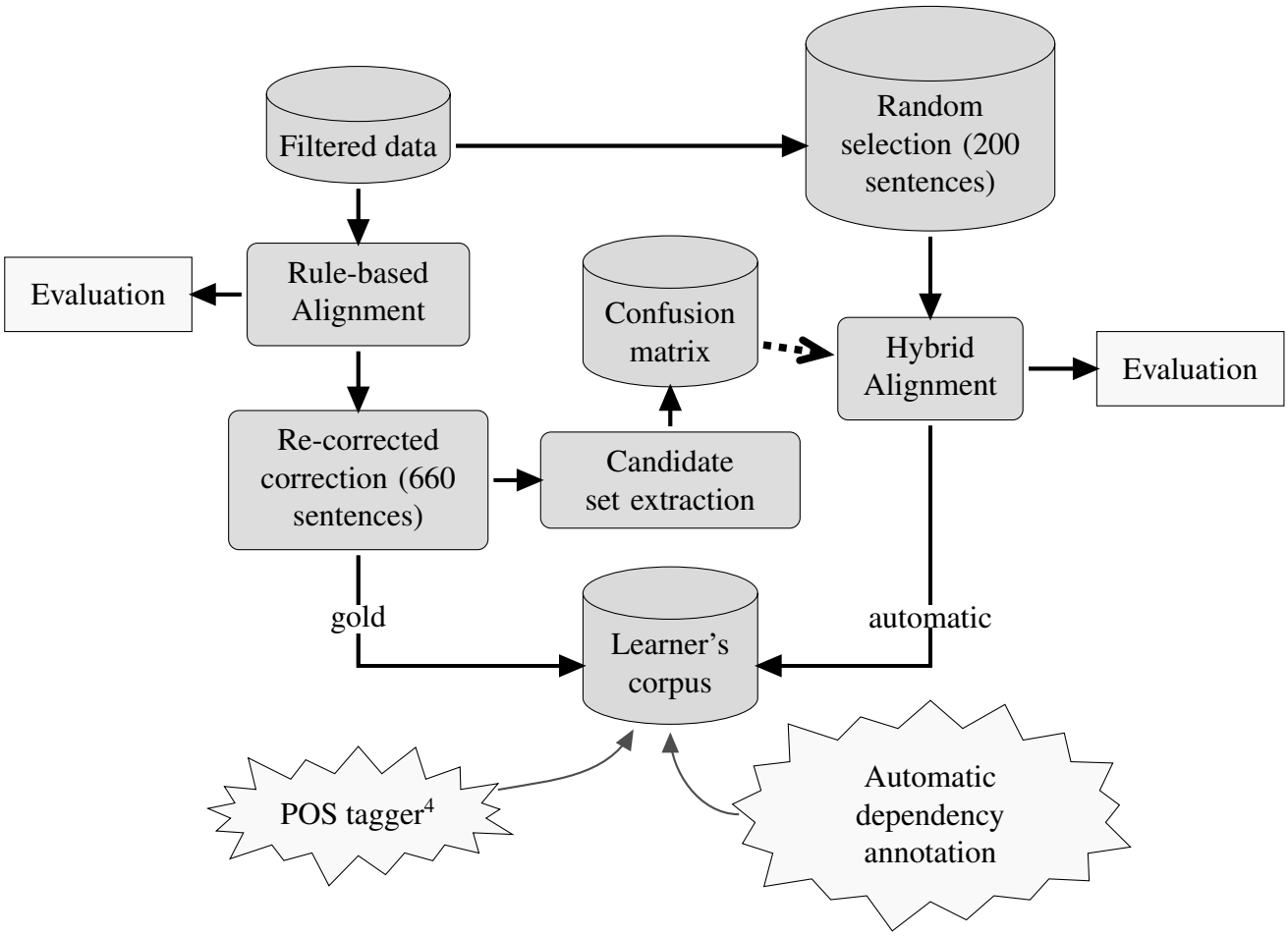


Figure 1: Semi automated alignment process

their word types, POS tag, word correction, and their error annotations.

4 Experiments

We conducted two experiments with this corpus. The first experiment was to show that re-corrected alignments help the automatic alignment. After cleaning noises, we split the data into two parts: 658 sentence pairs for *Rule-based* alignment and 5,557 sentence pairs for *Hybrid* alignment.

The second experiment had two goals. The first goal was to evaluate the difficulty of error identification of Indonesian learners" sentences. The second goal was to report the quality of human annotators in evaluating the automatic alignment results. Finally, in this section, we show how we used the corpus for preliminary error identification.

4.1 Semi Automatic Alignment

Figure 1 illustrates generally the semi automatic alignment. It was performed in two steps after filtered out the Lang-8 data as explained in Section 3. **First**, we did *Rule-based* alignment by constructing an *Edit Distance (ED)* matrix (Jurafsky and Martin, 2009) between a learner"s sentence and a native-corrected sentence using dynamic programming. Since the learner"s sentence and their correction are in the same language, we defined several heuristic rules to align them. Based on the ED matrix, we aligned a word from the learner"s sentence and a word from the native corrected sentence if they

[4] `http://septinalarasati.com/work/morphind/`

match each other. If the word does not match, we extracted the stem word (lemma) from morphology information obtained from Morphind (Larasati et al., 2011). If the stem words are match, we assigned the word pair as an *affixation* error (**A**). The second step, *Hybrid* alignment, is described later.

For pairs that do not have same stem word, we aligned them as *replacement error* (**R**) if their predecessor and their successor were also aligned. We called this the Neighboring Dependency Rule as explained in Wang and Zhou (2004) and shown in Equation 1. Let triple $\langle w_1, R, w_2 \rangle$ be a syntactic dependency relating two words w_1 and w_2 and a dependency relation R between them. Given a pair of learner sentences L and a corrected sentence C, where C is a native correction of L. L contains a triple $\langle l_1, R_L, l_2 \rangle$, and C contains a triple $\langle c_1, R_C, c_2 \rangle$.

$$align(R_L, R_C) \iff \begin{cases} align(l_1, c_1)\ and \\ align(l_2, c_2) \end{cases} \tag{1}$$

The method that was used in computing *spelling alignment* (**S**) in the *Rule-based* alignment is the Python built-in library `SequenceMatcher.ratio`[5] from `difflib` package. If the word pair follows the rule in Equation 2, the pair is annotated as a spelling error.

$$\begin{aligned} &abs(x_i.|vowel| - y_j.|vowel|) \leq 2\ \wedge abs(x_i.|consonant| - y_j.|consonant|) \leq 2 \wedge \\ &ratio \geq 0.7\ \implies\ (x_i, y_j)\ \textit{is a Spelling error} \end{aligned} \tag{2}$$

Depending on what rule they followed, an alignment was tagged as a spelling, affixation, or replacement errors. Next, the system annotated unaligned words in the learner"s sentence as unnecessary words and unaligned words in the corrected sentence as omitted words. After that, we asked a native speaker to evaluate and re-correct the alignment results again to create a gold standard alignment. Furthermore, we extracted a confusion matrix and computed the confusion matrix probability.

Second, we chose a candidate error from the confusion matrix in assigning a spelling, affixation, or replacement error in the *Hybrid* process based on Equation 3 to improve the precision before applying the same rules as in the *Rule-based* procedure to align the remaining words.

$$\hat{y_j} = arg \max_{y_j} M(y_j|x_i) \tag{3}$$

where M is the confusion matrix. In a different process, we performed POS tagging on every word using Morphind and merged them into the aligned sentences. We also manually assigned dependency relation for the Rule-based results, trained them using MST parser (McDonald et al., 2006), and ran an automatic parser for Hybrid results using the in-house training parser with 81.18% accuracy. This alignment was saved in XML format to achieve readability and easy extensibility. Table 2 shows the XML format for the annotation schema.

Figure 2 is the example of two alignments. For each example 2a and 2b, the upper sentence is a learner sentence and the below sentence is the native-corrected sentence. The vertical lines show alignments between two sentences; the curves represent the dependency relation. We provided two examples to show how one learner"s sentence was corrected in two different ways. The *Replacement* in Figure 2a and Figure 2b was corrected in the same way. Before we analyze the error, we will explain about clitic. The *clitic* '–nya' can be written together as one word after a transitive verb as a direct object, after an intransitive verb as an indirect object, after a noun as a determiner or third person possessive pronoun, or after a preposition as a preposition object. In this example, Morphind assigned 'nyanyinya' with VSA_PS3. However, our heuristic rule found that 'nyanyi' is an intransitive verb (VSA is changed into VSAI) that does not have a direct object. An intransitive verb can take a complement but no complement is written as a clitic. The conclusion is that the word 'nyanyinya' is an incorrect construction.

[5] `http://docs.python.org/2/library/difflib.html`

Attribute	Description
sentence_id	The id of the learner-corrected sentence
journal_id	The id of journal in which the sentence belongs to
learner_id	The id of learner related to learner's metadata
learner_L1	The id of learner's first language
learner_sentence	Sentence written by learner
lPos_tag	POS tag of learner's sentence
lDep_rel	Dependency relation of learner's sentence
corrected_sentence	Sentence corrected by native speaker
cPos_tag	POS tag of corrected sentence
cDep_rel	Dependency relation of corrected sentence
Token list	List of words that were corrected. Each token has index position in learner's sentence and corrected sentence, an error type, and a corrected word

Table 2: XML file structure

For Example 2a and 2b, the *relative pronoun* "yang" is never followed by a preposition; the word "yang" is followed by a verb or an adjective, so the words "yang di" is incorrect form. We use symbol λ for a null string. In this example, "yang di" can be corrected in two different ways. The first correction (2a) was to add a verb, "bermain", so the preposition "di" follows the verb and is labeled as a prep. The second correction (2b) was to delete the relative pronoun "yang", so the verb "mendengarkan" is directly followed by the preposition "di" and is labeled as a prep. Therefore, in Figure 2a, the object "adik" is in the room while in Figure 2b, the subject "Saya" is in the room.

4.2 Human Judgment

In the second experiment, we chose 100 sentences randomly and extracted sequences of five and seven words (partial sentence). We chose such limited contexts for humans to judge since those are the common window sizes used as features in machines learning. Then, we asked two native speakers to report three tasks: (TASK_A) is whether a partial sentence had an error, (TASK_B) is which word was incorrect, and (TASK_C) is what was their suggested correction. Finally, we computed the inter-annotator agreement using *Kappa Statistic* as $\kappa = \frac{P_a - P_e}{1 - P_e}$ (Chodorow et al., 2012) based on two first tasks.

4.3 Preliminary Error Identification

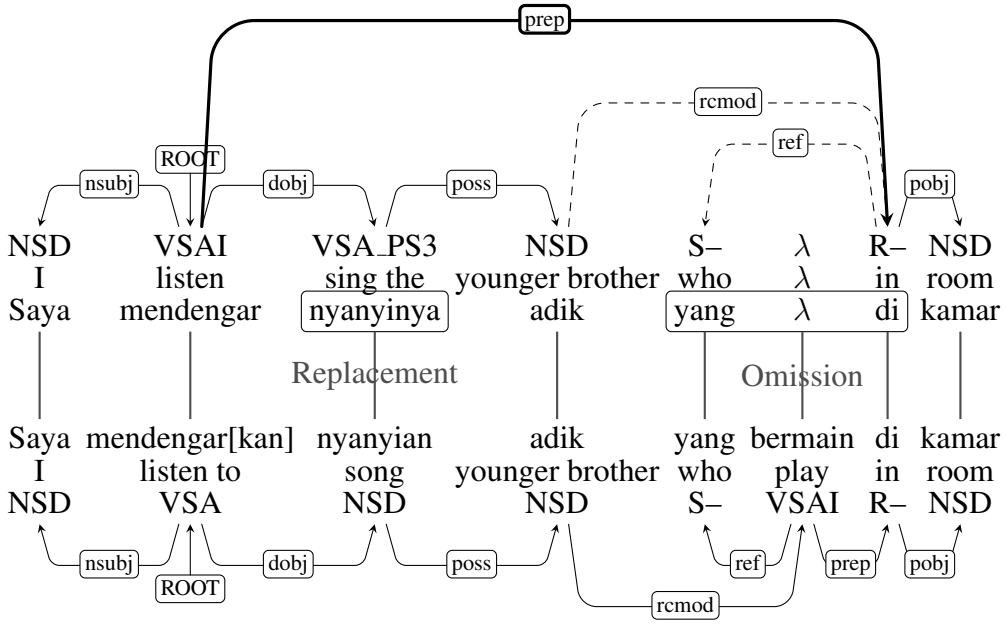
We utilized the corpus for preliminary experiment on learner's error identification using LibSVM[6] (Chang and Lin, 2011) with 10-fold cross-validation that run on the manual re-corrected alignment data. We used surface words, POSs, lemmas, prefixes, suffixes, biGrams. and triGrams as the features. We plan to use the corpus to develop an automatic system that gives a feedback to the learners about the mistakes that they make as well as the position of the mistaken words.

5 Evaluation

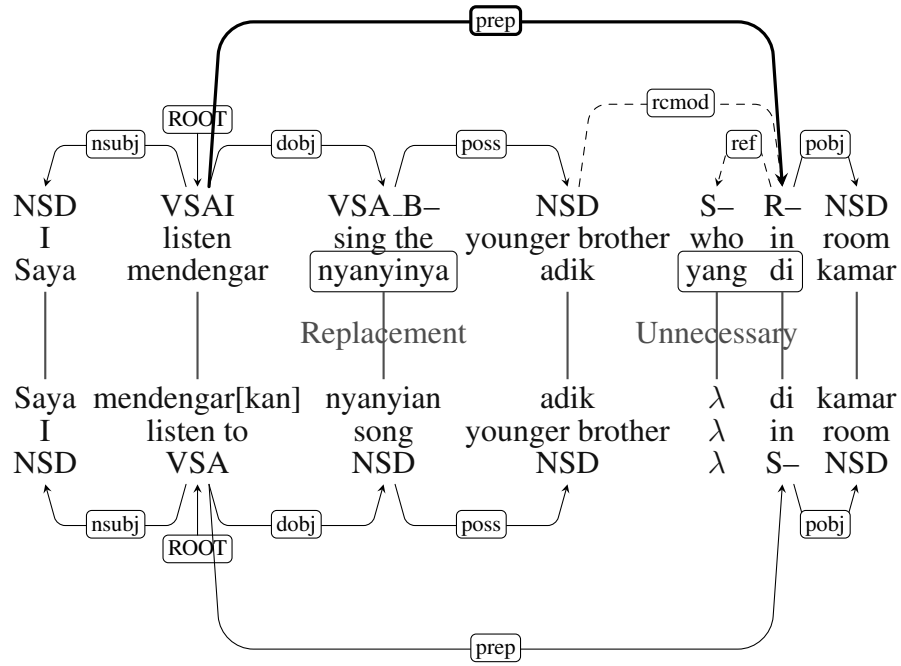
5.1 Evaluation on Semi Automatic Alignment

Based on the alignment results, Figure 3 shows that seven error types dominated the error distribution with more than 7%. We differentiated between OOV and X; OOV is all unknown words caused by a spelling error as well as words that are not used in writing while X is all words that are not

[6] http://www.csie.ntu.edu.tw/~cjlin/libsvm



(a) Replacement and omission example



(b) Replacement and Unnecessary Example

Figure 2: Error-corrected sentence alignment example

recognized by the POS tagger. Then we compared the *Rule-based* alignment results and native speaker re-corrected results.

To compare the accuracy, because we do not have a gold standard test data, we chose 202 random sentences and ran them on the *Rule-based* and *Hybrid* process. Then, we asked a native speaker to evaluate them manually. We computed the alignment precision by dividing the correct alignments by the number of automatic alignment results. In Figure 4, the precision are drawn as a white bar and a diagonal line bar. On average, the precision increased from 70.39% to 89.41%, and the sentence alignment completeness increased from 14.84% to 56.93%. A sentence was counted as complete if alignments and error types in one sentence are all correct. Unfortunately, the replacement precision for the Rule-based result is quite low because the system proposed more than one error annotation for each alignment to help manual evaluation, which means the number of error increased. We provided multiple annotations to help the annotator easily chose which error tag is appropriate for each

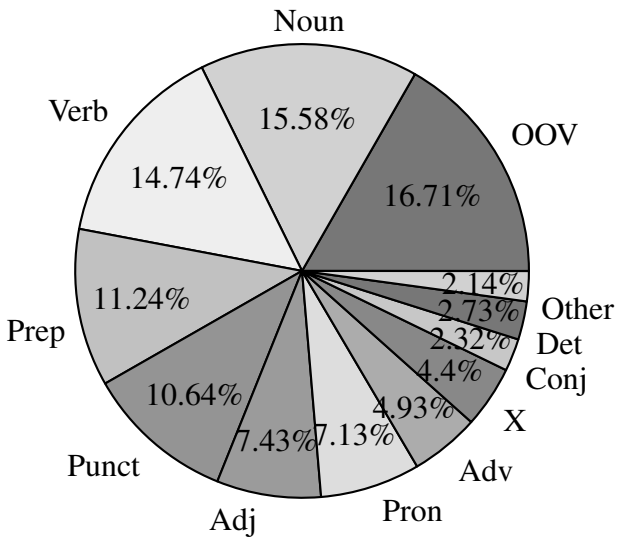


Figure 3: Error distribution

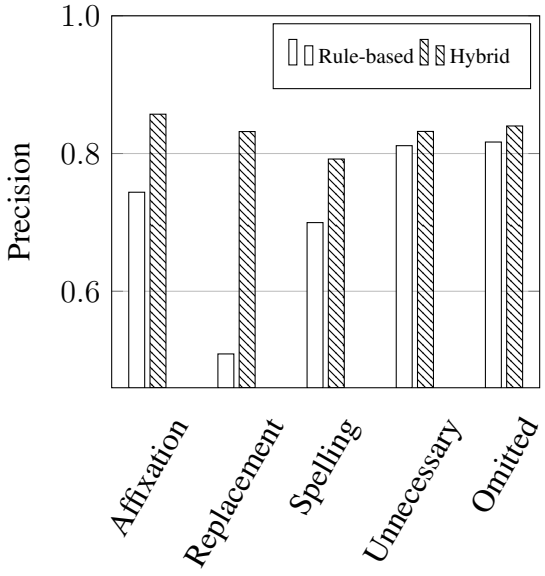


Figure 4: Precision for Rule-based and Hybrid alignment

Machine Annotation		Corrected Annotation					Multi Alignment
		A	R	S	U	M	
Affixation	A	6	0	0	1	0	1
Replacement	R	1	88	1	14	4	0
Spelling	S	17♣	3	80	0	0	1
Unnecessary words	U	1	16*	3	114	0	3
Omission words	M	0	24**	0	0	126	0

Table 3: Confusion matrix of error-corrected sentences from randomly selected 202 sentences

alignment.

This experiment shows that the *confusion matrix* from manual error annotation improves the alignment quality of the learner's error-corrected corpus. Table 3 represents an analysis of the confusion matrix. Figure 4 indicates that the automated system aligns word pairs well, but Table 3 shows that the system cannot assign a correct error type for some cases. For example, (1) affixation errors that were identified as spelling error (♣) (if the correction words were only one or two characters, similar to one of the affixes, and were unavailable in the confusion matrix); (2) some replacements were identified as unnecessary words(*) or omitted words (**); and (3), incorrect replacements happen because the *Neighboring Dependency Rule* failed to identify them. In Table 3, some spelling errors were identified as affixation errors because the system cannot differentiate spelling and affixation errors if the word pair satisfy both spelling and affixation rules. The morphology analyzer, Morphind, cannot produce a correct affixation if the word is spelled incorrectly. This table also shows some unnecessary words and omitted words were identified as replacement. The unnecessary words and omitted words were miss-identified because a native speaker made a correction that changed the order of the sentence so that they satisfied the *Neighboring Dependency Rule*.

Figure 5 shows The result of the alignment example from Figure 2 in in XML format. Example 2a and 2b are represented as XML with sentence id "`5`" and "`6`" respectively for the same journal id (`jId`), learner L1 (`nId`), and user id (`uId`). We use same variables as were listed in Table 2. The tag s represents a sentence where tag t represents a token. We specified the omitted error type differently, which is tagged as **I** that means 'insert'. The dependency relation for each word is written as `head:label` while the affix is written as `prefix|suffix`. A word that does not have a prefix or suffix is written as -|-. The omitted word still has i variable although the corresponding word is

```
...
<s> id=  5   jId="598062   nId="5" uId="179855"
oText= S a y a  m e n d e n g a r  n y a n y i n y a  a d i k  y a n g  d i  k a m a r  oPOS= P S 1  V S A  V S A I _ P S 3  S -- R N S D
     oAffixes=  -|- m e n |- -|n y a -|-
-|- -|- -|-   oStem= s a y a  d e n g a r  n y a n y i  a d i k  y a n g  d i  k a m a r  oDeps= 1 : n s u b j  -1:root 1
    :dobj 2:poss 5:ref 3:rcmod 5 : p o b j
cText= S a y a  m e n d e n g a r k a n  n y a n y i a n  a d i k  y a n g  b e r m a i n  d i  k a m a r  oPOS= P S 1  V S A  N S A  N S D
     S V S A  R N S D  cAffixes=  -
|- m e n |k a n -|- -|- -|- b e r |- -|- -|-   cStem= s a y a  d e n g a r  n y a n y i  a d i k  y a n g  m a i n  d i
     k a m a r  cDeps= 1 : n s u b j  -1:root 1dobj 2:poss 5:ref
3:rcmod 5:prep 6 : p o b j
<t i=  2   j=  2   p= n y a n y i n y a ; n y a n y i a n  e T=  R  >nyanyinya</t>
<I i=  5   j=  5   c= b e r m a i n  >
</s>
<s> id=  6   jId="598062   nId="5" uId="179855"
oText= S a y a  m e n d e n g a r  n y a n y i n y a  a d i k  y a n g  d i  k a m a r  oPOS= P S 1  V S A  V S A I _ P S 3  S -- R N S D
     oAffixes=  -|- m e n |- -|n y a -|-
-|- -|- -|- -|-   oStem= s a y a  d e n g a r  n y a n y i  a d i k  y a n g  d i  k a m a r  oDeps= 1 : n s u b j  -1:root
    1:dobj 2:poss 5:ref 3:rcmod 5 : p o b j
cText= S a y a  m e n d e n g a r k a n  n y a n y i a n  a d i k  d i  k a m a r  oPOS= P S 1  V S A  N S A  N S D  R N S D  Z --
    cAffixes=  -|- m e n |k a n -|- -|- -|- -
|- -|-   cStem= s a y a  d e n g a r  n y a n y i  a d i k  d i  k a m a r  cDeps= 1 : n s u b j  -1:root 1:dobj 3:poss
    1:prep 5 : p o b j
<t i=  2   j=  2   p= n y a n y i n y a ; n y a n y i a n  e T=  R  >nyanyinya</t>
<t i=  4   j=  4   eT=  D  >yang</t>
</s>
...
```

Figure 5: Alignment result in XML format

Agreement	5-words	7-words
TASK_A (κ)	0.7067	0.8652
TASK_B (κ)	0.5333	0.8333

Table 4: Inter-annotator Agreement of Partial Sentence Error Identification

not available in the learner"s sentence to identify where the word is inserted. For the same reason, for an unnecessary word, the j variable has the value of the index of the next word after deletion. The error tag for these examples are translated to *verb replacement* (**RV**), *verb insertion* (**IV**) for sentence `id=5`; and *verb replacement* (**RV**), *conjunction omitted* (**OS**) for sentence `id=6`. The POS, the stem word, and the dependency relation are written as a sentence element because we only save the token that has a correction. For easy used by any researcher, the automatic result of the corpus, without the dependency construction, can be accessed freely from `http://sourceforge.net`[7].

5.2 Evaluation on Human Judgment

Table 4 shows the κ score for TASK_A and TASK_B defined in Section 4.2. In general, a 7-word context performs better agreement while the 5-words partial sentences are more likely to be mistaken. For TASK_A the native annotators have better agreement on identifying whether the partial sentence has an error compared to the TASK_B of identifying which word is grammatically erroneous. The TASK_B shows that the native speakers sometimes disagree about which word is incorrect, typically

[7] `http://sourceforge.net/projects/indonesianlearnercorpus/files/IndLCorpus/IndLearnerCorpus.xml`

Kinds of error	5-words	7-words
(1) Unidentified	2	3
(2) Error detection mismatch	15	4
(3) Error correction mismatch	17	7

Table 5: Error classification based on Human Judgment

Error Types	Classification	
	Precision	Recall
*Replacement	**0.0891**	**0.5769**
After splitting error type		
Noun	0.1600	0.8000
Verb	0.1034	0.3158
Preposition	0.3402	0.6735
Adjective	0.3139	0.7917
Average Replacement	**0.2293**	**0.6452**

Table 6: Classification based result for error detection system

in grammatical cases. This disagreement is because human possibly pick different word as en error in same sentence. This indicates by the lowest agreement score in Table 4 for 5-word column.

In addition, we defined three categories to classify human judgment: (1) "Unidentified", (2) "Error detection mismatch", and (3) "Error correction mismatch". We organized the human judgments into these three categories and show the result in Table 5. In category (1), unidentified, the annotator was confused about the meaning of the partial sentence and was not able to make a decision. In category (2), error detection mismatch, one annotator stated that the sentences had an error while the other annotator said no error. In fact, such sentences were acceptable because at least one native classified it as correct. Category (3), error correction mismatch, was identified if the sentence had an error, but the annotators identified different word as the error so they proposed different corrections. The last category also supports the conclusions deduced form Table 4 that identifying the word containing the error sometime depend on human perception. The higher score of inter-annotator agreement in Table 4 and decrease in mismatches in Table 5 show that in longer context, humans have better agreed upon sequence of word. For the complete sentence we got *Kappa Statistic* κ about 0.8955.

As preliminary work on automatic error identification, these experiment showed that some error types such as replacements, affixations, and omitted words were difficult to identify. To get better accuracy, we classified these errors into a more fine-grain error type based on the POS information. We subdivided the affixation errors into inflection and derivation errors, which is split into noun and verb inflections and derivations. We divided replacement errors into several errors based on their POS. We found that splitting the replacement errors into more fine-grain error types improved the accuracy. As shown in Table 6, for example, we report the results of the replacement error type detected by our model using the SVM classifier. This table also shows the score of more fine-grain error types as nouns, verbs, prepositions, and adjectives. *Average Replacement* is computed from the average score of noun, verb, preposition, and adjective for each precision and recall. This table shows that for the four error types, the average precision and recall improve 14.02% and 6.83% respectively from the single run replacement (*Replacement, the first line in the table). We will use the replacement result as a baseline for our next experiment. The model indeed needs to be improved significantly by extracting more features such as the dependency features and information that can be generated from a large native corpus as well as run on more sophisticated method.

As a starting point of providing learner resource of Indonesian language, a confusion matrix can be extracted from this corpus to help language teacher, for instance, to conceive the more frequent learner errors, which is not made by native speaker and suggest destructive answer in creating multiple choice question answering for evaluation of learner ability.

6 Conclusion

We present our work about developing a corpus of second language learners. Our work is the first effort in creating a learner corpus for Indonesian language which is one of the under develop languages. We show how we organized the error annotation information for each sentence in XML format and

how we improved the automatic error alignment by extracting confusion matrix that help in assigning the error type for each alignment.

In the second experiment we verify that semantic errors in a partial sentence are difficult to judge even by humans. In Indonesian language, there is a possibility that a sentence is written in several ways because of its free word ordering. Specifically, it is natural to have more than one correction for one grammatically wrong sentence and supposedly to use longer context to work with statistic method or machine learning. Indeed, the annotator agreement for complete sentence also represented the quality of the manual check of the alignments.

Lastly, we also show our preliminary stage of utilizing the corpus and found that this corpus can be used in error identification of learner sentences. Our next work is to extend the error identification using syntactic information combines dependency rules extracted from normal sentences. We also want to utilize a large native corpus to get more information to enrich our feature set.

For emerge direction, we will utilize the corpus to develop error-detection system to show the error type and error position as the feedback for the second language learners and as resource for developing other NLP tools for Indonesian such as dependency parser, name entity, or phrase segmentation. On the other hand, this resource is also useful for language teacher as describe in the previous section.

Acknowledgments

This study is supported in part by the Directorate General of Higher Education, Republic of Indonesia under BPPLN Scholarship Batch 7 fiscal year 2012-2015. The authors would like to thank the unknown reviewer, and Mike Barker, and Lis Kanashiro for valuable comments.

References

Abuhakema, G., Faraj, R., Feldman, A., and Fitzpatrick, E. (2008). Annotating an Arabic Learner Corpus for Error. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Alwi, H. (2000). *Tata Bahasa Baku Bahasa Indonesia*. Balai Pustaka, Indonesia, third edition.

Boyd, A. (2010). EAGLE: an Error-Annotated Corpus of Beginning Learner German. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC10)*, pages 1897–1902, Valletta, Malta.

Chang, C.-C. and Lin, C.-J. (2011). LIBSVM: A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27.

Chodorow, M., Dickinson, M., Israel, R., and Tetreault, J. R. (2012). Problems in Evaluating Grammatical Error Detection Systems. In *COLING*, pages 611–628. Indian Institute of Technology Bombay.

Dahlmeier, D., Ng, H. T., and Wu, S. M. (2013). Building a large annotated corpus of learner english: The nus corpus of learner english. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 22–31, Atlanta, Georgia. Association for Computational Linguistics.

Fraser, A. and Marcu, D. (2006). Semi-Supervised Training for Statistical Word Alignment. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 769–776, Sydney, Australia.

Hana, J., Rosen, A., Škodová, S., and Štindlova, B. (2010). Error-tagged Learner Corpus of Czech. In *Proceedings of the Fourth Linguistic Annotation Workshop, ACL*, pages 11–19, Upsala, Sweden. Association for Computational Linguistics.

Izumi, E., Uchimoto, K., and Isahara, H. (2005). Error Annotation for Corpus of Japanese learner English. In *Proceedings of the Sixth International Workshop on Linguistically Interpreted Corpora (pp. 71-80). Jeju Island, Korea: Association for Computational Linguistics*.

Jurafsky, D. and Martin, J. H. (2009). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice-Hall, New Jersey, second edition.

Larasati, S. D. (2012). IDENTIC Corpus: Morphologically Enriched Indonesian-English Parallel Corpus. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC12)*, pages 902–906, Istanbul, Turkey.

Larasati, S. D., Kuboň, V., and Zeman, D. (2011). Indonesian Morphology Tool (MorphInd): Towards an Indonesian Corpus. In *SFCM*, volume 100 of *Communications in Computer and Information Science*, pages 119–129, Zurich, Switzerland.

Leacock, C., Chodorow, M., Gamon, M., and Tetreault, J. (2014). *Automated Grammatical Error Detection for Language Learners*. Morgan and Claypool Publishers.

Lee, S.-H., Dickinson, M., and Israel, R. (2012). Developing Learner Corpus Annotation for Korean Particle Errors. In *Proceedings of the Sixth Linguistic Annotation Workshop*, LAW VI '12, pages 129–133, Stroudsburg, PA, USA. Association for Computational Linguistics.

Manurung, R., Distiawan, B., and Putra, D. D. (2010). Developing an Online Indonesian Corpora Repository. In *Proceedings of the 24th Pacific Asia Conference on Language, Information, and Computation*, pages 243–249, Sendai, Japan.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of english: The penn treebank. *Comput. Linguist.*, 19(2):313–330.

McDonald, R., Lerman, K., and Pereira, F. (2006). Multilingual Dependency Analysis with a Two-stage Discriminative Parser. In *Proceedings of the Conference on Computational Natural Language Learning (CONLL)*, pages 216–220.

Nagata, R., Whittaker, E. W. D., and Sheinman, V. (2011). Creating a Manually Error-Tagged and Shallow-Parsed Learner Corpus. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pages 1210–1219, Portland, Oregon, USA.

Ng, H. T., Wu, S. M., Wu, Y., Hadiwinoto, C., and Tetreault, J. (2013). The CoNLL-2013 Shared Task on Grammatical Error Correction. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–12, Sofia, Bulgaria. Association for Computational Linguistics.

Rozovskaya, A. and Roth, D. (2010). Annotating ESL Errors: Challenges and Rewards. In *Proceedings of the NAACL HLT 2010 Fifth Workshop on Innovative Use of NLP for Building Educational Applications*, IUNLPBEA '10, pages 28–36, Stroudsburg, PA, USA. Association for Computational Linguistics.

Sneddon, J. N., Adelaar, A., Djenar, D. N., and Ewing, M. C. (2010). *Indonesian: A Comprehensive Grammar*. Roultedge, Australia, second edition.

Wang, W. and Zhou, M. (2004). Improving Word Alignment Models Using Structured Monolingual Corpora. In *Proceedings of Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pages 198–205, Barcelona, Spain.