OPEN ACCESS
Indo-JC
Indonesia Journal of Computing
ISSN 2460-9056
socj.telkomuniversity.ac.id/indojc

Ind. Journal on Computing
Vol. 8, Issue. 3, December 2023. pp. 1-10
doi:10.34818/indojc.2023.8.3.709

Generating Indonesian Paraphrased Sentences with Verbal Predicate Replacement

Bunyamin #1, Arie Ardiyanti Suryani #2

# Natural Language Processing and Text Mining Laboratory,
School of Computing, Telkom University, Bandung, Indonesia
[1] bunbunyamin@telkomuniversity.ac.id
[2] arieardiyanti@ telkomuniversity.ac.id

Abstract

Paraphrasing a sentence is restating a sentence using different diction without changing the meaning of the language. Paraphrasing is generally done by writers to avoid plagiarism or to make paraphrased sentences easier to understand. Paraphrase generation is also useful for developing Natural Language Processing applications in question and answering, linguistic-based fields of stenography, recommender systems, or machine translation. Paraphrasing a sentence can be done in several ways, including using synonymous substitution techniques, changing the form of the sentence, or changing the predicate part of the sentence. The paraphrasing carried out in this research is identifying the type of verb predicate in a simple sentence using PoS Tagging, then looking for words that are similar to the predicate using the word2vec language model. The antonym list is used to correct the substitution results. Evaluation is carried out using human judgment which compares the resulting sentence and the original sentence. The experimental results show that of the 600 simple sentence dataset, 48.37% of the sentences have semantic similarity, 20.93% have semantic reduction, and 30.70% have no semantic similarity.

Keywords: Paraphrase, Predicate, Semantic Similarity, word2vec

Abstrak

Parafrase kalimat adalah menyatakan kembali kalimat dengan menggunakan diksi yang berbeda tanpa mengubah makna bahasa. Parafrase umumnya dilakukan oleh penulis agar terhindar dari plagiarism atau agar kalimat hasil parafrase menjadi lebih mudah dimengerti. Pembangkitan parafrase berguna juga untuk pembangunan aplikasi Pemrosesan Bahasa Alami dalam tanya jawab, bidang stenografi berbasis linguistik, sistem perekomendasi, atau mesin penerjemah. Parafrase kalimat dapat dilakukan dengan beberapa cara, antara lain dengan teknik substitusi sinonim, mengubah bentuk kalimat, atau mengganti bagian predikat kalimat. Parafrase yang dilakukan dalam penelitian ini adalah mengidentifikasi jenis predikat verba dalam kalimat sederhana menggunakan PoS Tagging, kemudian mencari kata yang mirip predikatnya menggunakan model bahasa word2vec. Daftar antonim digunakan untuk memperbaiki hasil substitusi. Evaluasi dilakukan dengan menggunakan penilaian manusia yang membandingkan kalimat hasil dan kalimat asalnya. Hasil percobaan menunjukkan bahwa dari 600 dataset kalimat sederhana, 48,37% kalimat memiliki kesamaan semantik, 20,93% mengalami reduksi semantik, dan 30,70% tidak memiliki kesamaan semantik.

Kata Kunci: Parafrase, Predikat, Kesamaan Semantik, word2vec

Received on September, 2023. Accepted on December, 2023

I. Introduction

Paraphrasing is the re-expression of a speech from one level or type of language into another speech without changing the meaning [1]. Paraphrasing sentences have similar meanings but are written in different sentences [2]. Paraphrasing sentences do not always have the same meaning synonyms but can be broader, in a longer or shorter form, with more or less the same meaning as the original sentence. For example, the statement *Kami tinggal di Bandung* can be paraphrased as *Kami menetap di Bandung* or *Bandung adalah kota tempat tinggal kami*.

Paraphrasing is generally carried out by writers in citation activities in writing scientific papers. Apart from making the paraphrased sentences easier to understand, the cited sentences used must be different from the original to avoid plagiarism. A paraphrase sentence generator will therefore be very useful.

Sentence paraphrasing can be divided into three research focuses, namely identifying paraphrases between two parts of the text, generating paraphrases if one input text is known, and paraphrasing extraction, which contains essential things from a part of the text [3]. In terms of identifying paraphrasing sentences, it can be done to complete the task of knowing the semantic similarities between two parts of the sentence or grouping topics/documents. The generation of sentence paraphrases can be applied to solve sentence formation problems, change sentence style [4], change the sentence level of natural sentences [5], and improve results and translation. Meanwhile, in terms of paraphrasing extraction, it can be used for text summarization and text simplification.

Many paraphrasing is done to help research in various areas, paraphrasing literary sentences William Shakespeare's texts in informal language colloquial language, the technique used is a machine translation approach [3]. Paraphrasing applications are also carried out on the task of changing an unnatural speech language Unsuitable Expression for Spoken Language UES into a more natural speech-language Suitable Expression for Spoken language SES [4]. Kaji *et al.* examined paraphrasing by focusing on connotational differences, namely from the style of language stylistic or the formality of the language [5]. The technique used is to distinguish UES and SES based on the co-occurrence probability between the Written and Spoken language of the corpus collected from the web. This technique produces a paraphrasing accuracy of 75-76% using 240 web pages consisting of 6.1M predicates from the written corpus and 11.7M from the spoken language corpus. Research on paraphrasing important sentences can be applied to other research areas, such as question answering [6] [7], machine translation [8] [9] [10], semantic parsing [11] [12], and data augmentation [13].

Another research in paraphrasing was research conducted by Barmawi and Muhammad which also paraphrases the lexical replacement technique and uses a syntactic parser to get the context of the lexical to be replaced [14]. This paper gives excellent results above the baseline lexical substitution without including context.

In this research, authors will replace verb-type predicates in a simple sentence using a language model built using word2vec. The corpus to be used for model development comes from the Indonesian language wikipedia. To improve the lexical replacement process, Indonesian language stemmers and antonym word lists are used. Thus, a paraphrase generator for simple sentences in Indonesian is obtained.

II. Research Method

The main idea in this study is to utilize a language model built using word2vec. The word to be replaced is calculated for its similarity by using the *most similar* function to the existing vocabulary in the model. After getting candidate words that are similar, then it will be chosen which one has the highest similarity value but not the one that has the opposite meaning found in the antonym word list. In order to be able to use the antonym word list, a stemmer is used, to change the word to be replaced and the candidate word to become a root word. The stages of the proposed research are shown as Fig. 1. Meanwhile the algorithm is shown in the Algorithm 1.

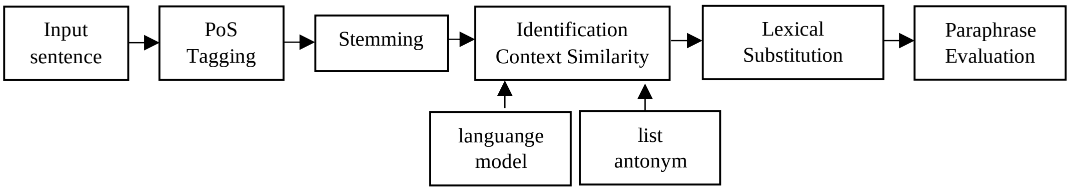


Fig. 1. Research method

ALGORITHM 1: Indonesian Paraphrased with Verbal Replacement

```
Input: simple sentence
Output: Paraphrased simple sentence
import postagging, ndetc_stemmer, word2vec
input(simple_sentence)
tokens ← tokenize(simple_sentence)
postag_result ← postagging(tokens)
paraphrased ← copy(postag_result)
for i in postag_result do
      if postag_result[0][i][1] is "VB" then
          if not postag_result[0][i][0] in vocab_model then
              para ← (postag_result[0][i], "VB")
          else
              para ← most_similar(postag_result[0][i][0], topn=15)
          endif
          root_1 ← ndetc_stemmer(postag_result[0][i][0])
          root_2 ← empty_list
          for j in len(para) do
              root_2.append([para[j][0], ndetc_stemmer([para[j][0])
          endfor
          idx ← 0
          same ← false
          while idx < len(root_2) and not same do
              different_word ← root_1 is not root_2[idx][1]
              antonym_word ← checkAntonym(root_1, root_2[idx][1])
              if different_word and not antonym_word then
                  paraphrased[0][1] ← (root_2[idx][0], "VB")
                  same ← true
              else
                  idx ← idx + 1
              endif
          endwhile
      endif
endfor
output(paraphrased)
```

A. Input Sentences

The input sentences used in this study are simple sentences. A simple sentence is a sentence that is formed from a clause whose elements are simple words or phrases. A simple sentence structure consists of a subject, predicate, object, and adverb. The patterns found in simple sentences are: a. Subject + Predicate, for example: “Saya makan”. b. Subject + Predicate + Object, for example: “Saya makan nasi”, c. Subject + Predicate +

Object + Adverd, for example: “Kamu minum air di rumah”, “Adik bermain bola kemarin”, d. Subject + Predicate + Object + Object, for example: “Saya memberikan hadiah kepada adik”.

B. *Part of Speech (PoS) Tagging*

Part of Speech Tagging is the process of marking word classes in input sentences such as verbs, nouns, adjectives, and adverbs (Table 1). One simple sentence consists of one predicate verb. With this tagging process, the verb-type predicate part of simple sentences is identified. In this study, Indonesian language PoS Tagger was used as a result of training using CRF Tagger from nltk [15].

TABLE I
Part of Speech Tagger’s Type

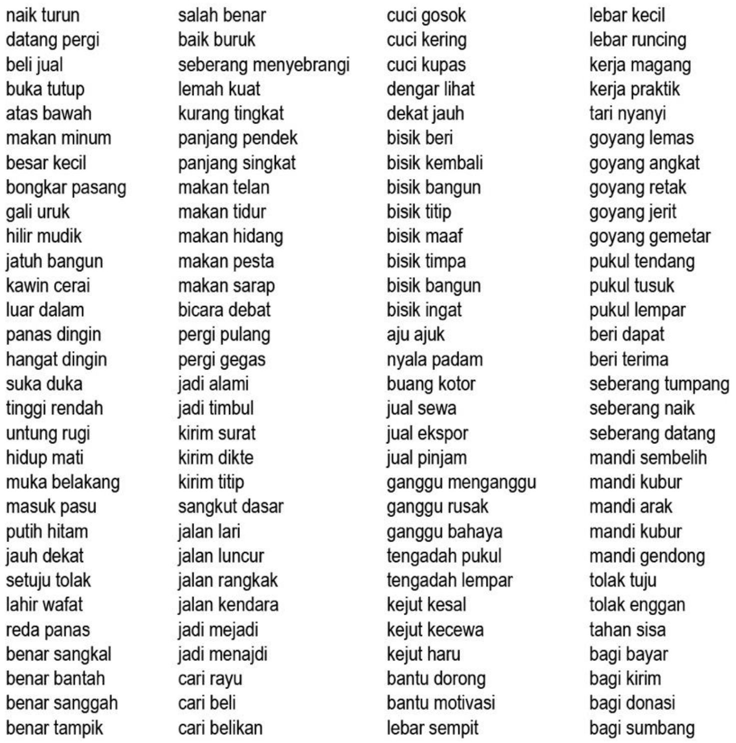
Tagger Type	Meaning	Sample
CC	coordinating conjuction	“lalu”
CD	cardinal digit	“setiap”, “kedua”, “satu”
DT	determiner	“para”
EX	foreigner word	“table”
IN	preposition/subordinating conjunction	“dalam”, “oleh”, “di”
JJ	adjective	“baik”
JJR	adjective, comparative	-
JJS	adjective, superlative	-
LS	list marker 1	-
MD	modal	“sedang”, “akan”, “harus”
NN	noun, singular	“orang”
NNS	noun plural	-
NNP	proper noun, singular	“ayah”
NNPS	proper noun, plural	-
PDT	predeterminer	-
POS	possessive ending	-
PRP	personal pronoun	“dia”, “saya”, “mereka”
PRP$	possessive pronoun	“ia”
RB	adverb	“juga”, “pasti”, “itu”
RBR	adverb, comparative	-
RBS	adverb, superlative	-
RP	particle	-
TO	to go	-
UH	interjection	-
VB	verb, base form	“mengetahui”, “menempel”
VBD	verb, past tense	-
VBG	verb, gerund/present participle	-
VBN	verb, past participle	-
VBZ	verb, 3rd person sing. present	-
WDT	wh-determiner	-
WP	wh-pronoun	-
WP$	possessive wh-pronoun	-
WRB	wh-abverd	-

C. Stemming

Stemming is used to find the root word of the verb predicate (VB). After finding the root word, this root word will be compared with the root word of each word candidate generated by the word2vec model to check whether the word is included in the antonym. The NDETC stemmer is used for stemming in the proposed method [16].

D. Identify Context Similarities

This process determines the word candidates from language model that have a similar meaning with the verb-type predicate from input sentence. The process of identifying the similarity of this context will use language model from word2vec [18]. Word2vec has shortcomings in issuing words that have similarities, for example the word “beli” has a high similarity with “jual”, even though the semantical meaning is opposite. To overcome this problem, a list of opposite words antonyms is developed to filter the wrong words (Fig. 2). The list of antonyms was compiled based on the results of preliminary experiments. The number can continue to be increased to overcome weaknesses in word2vec.



naik turun
datang pergi
beli jual
buka tutup
atas bawah
makan minum
besar kecil
bongkar pasang
gali uruk
hilir mudik
jatuh bangun
kawin cerai
luar dalam
panas dingin
hangat dingin
suka duka
tinggi rendah
untung rugi
hidup mati
muka belakang
masuk pasu
putih hitam
jauh dekat
setuju tolak
lahir wafat
reda panas
benar sangkal
benar bantah
benar sanggah
benar tampik

salah benar
baik buruk
seberang menyebrangi
lemah kuat
kurang tingkat
panjang pendek
panjang singkat
makan telan
makan tidur
makan hidang
makan pesta
makan sarap
bicara debat
pergi pulang
pergi gegas
jadi alami
jadi timbul
kirim surat
kirim dikte
kirim titip
sangkut dasar
jalan lari
jalan luncur
jalan rangkak
jalan kendara
jadi mejadi
jadi menajdi
cari rayu
cari beli
cari belikan

cuci gosok
cuci kering
cuci kupas
dengar lihat
dekat jauh
bisik beri
bisik kembali
bisik bangun
bisik titip
bisik maaf
bisik timpa
bisik bangun
bisik ingat
aju ajuk
nyala padam
buang kotor
jual sewa
jual ekspor
jual pinjam
ganggu menganggu
ganggu rusak
ganggu bahaya
tengadah pukul
tengadah lempar
kejut kesal
kejut kecewa
kejut haru
bantu dorong
bantu motivasi
lebar sempit

lebar kecil
lebar runcing
kerja magang
kerja praktik
tari nyanyi
goyang lemas
goyang angkat
goyang retak
goyang jerit
goyang gemetar
pukul tendang
pukul tusuk
pukul lempar
beri dapat
beri terima
seberang tumpang
seberang naik
seberang datang
mandi sembelih
mandi kubur
mandi arak
mandi kubur
mandi gendong
tolak tuju
tolak enggan
tahan sisa
bagi bayar
bagi kirim
bagi donasi
bagi sumbang

Fig. 2. List of antonyms.

The list of antonyms in Fig. 2 is arranged not based only on the contrary meaning but according to the experimental results. The word "masuk" is paired with "pasu", because there are the words "memasukkan" and "memasukan" in the vocabulary in the word2vec model as a result of the training. Vocabulary "memasukan" is generated due to writing errors "memasukkan". When searched for the root word will produce the words "masuk and "pasu". They have very close values in similarity but different meanings. Another example is the word "benar" true which is paired with the words “sangkal”, "bantah”, “sanggah”, “tampik”, because there is the word "membenarkan" which is close to "menyangkal", "membantah", “menyangkal", and "menampik” but have different meanings. Also, the word “mandi” is paired with “kubur” because there are words “memandikan” and “menguburkan” which are close in similarity value but have different meanings.

E. Lexical Substitution

Lexical substitution is the process of replacing a predicate with a predicate that has the best similarity in the context of the sentence from the previous process, after filtering by using antonym list.

F. Paraphrase Evaluation

The paraphrase results are evaluated by expert judgment to assess the semantic closeness of the sentence and the similarity of the paraphrased sentence to the original sentence. The purpose of paraphrasing is to generate different sentences with similar or even the same meaning, and then the lexical similarity will be measured. A value of 0 from the evaluator stated that the output sentence had a different paraphrase in the semantic meaning of the input sentence, and a value of 1 stated that the semantic meaning of the output sentence had some meaning in the input sentence. In contrast, a value of 2 stated that the semantic meaning of the output sentence had the same meaning as the input sentence.

Table 2 shows the sample process of the research method. Input in the form of sentences is tokenized and post-tagged to identify words of type VB. For the word of type VB earlier, it is stemmed to get the root word. For example, 'memakan' is a word of type VB and the stemming result is ‘makan'. In the list of antonyms, the word ‘makan’ has the opposites ‘telan’, ‘tidur, ‘hidang’, ‘pesta’, and ‘sarap’. Thus, the word ‘telan’ is not chosen as a substitute word, but the word ‘mengkonsumsi’ is chosen.

TABLE 2
The Sample Process

No	Process	Result
1	Input sentences	Saya memakan nasi
2	Tokenization	['Saya', 'memakan', ‘nasi’]
3	PoS Tagging	[[('Saya', 'PRP'), ('memakan', 'VB'), ('nasi', 'NN')]]
4	Stemming	makan
5	Identification Context Similarities	[['menelan', 'telan'], ['mengkonsumsi', 'konsumsi'], ['mengonsumsi', 'konsumsi'], ['memangsa', 'mangsa'], ['menyantap', 'santap'], ['memakannya', 'makan'], ['memuntahkan', 'muntah'], ['dimakan', 'makan'], ['menghisap', 'menghisap'], ['mengisap', 'isap'], ['membutuhkan', 'butuh'], ['menghabiskan', 'habis'], ['meminum', 'minum'], ['mengunyah', 'kunyah'], ['membuang', 'buang']]
6	Lexical Substitution	[[('Saya', 'PRP'), ('mengkonsumsi', 'VB'), ('nasi', 'NN')]]
7	Output	Saya mengkonsumsi nasi

III. Results And Discussion

Six hundred simple sentence datasets derived from novels and news have been experimented with as input to the system built. The datasets were examined by five evaluators with Indonesian language expertise. Table 3 and Fig. 3 show that the second category has 48.83% average score, the first category has 21.50% average score, and the zero category has 29.67% average score. From these results it can be concluded that the system created has a good score for the assessment of 5 evaluators.

Correlation analysis using Rank Spearman Correlation shows that all correlation coefficient evaluators range from 0.40326 to 0.59525 (Table 4). This means that among the evaluators there is a correlation between their evaluation result, even though the correlation is not that strong. The evaluators 4 and 5 have the highest correlation coefficient of 0.59525, meawhile the evaluators 1 and 3 have the lowest one 0.40326.

TABLE 3
THE EVALUATION RESULT

Category	Evaluator 1	Evaluator 2	Evaluator 3	Evaluator 4	Evaluator 5	Average
0	255	75	122	260	209	30.70%
1	52	233	132	99	112	20.93%
2	293	292	346	241	279	48.37%

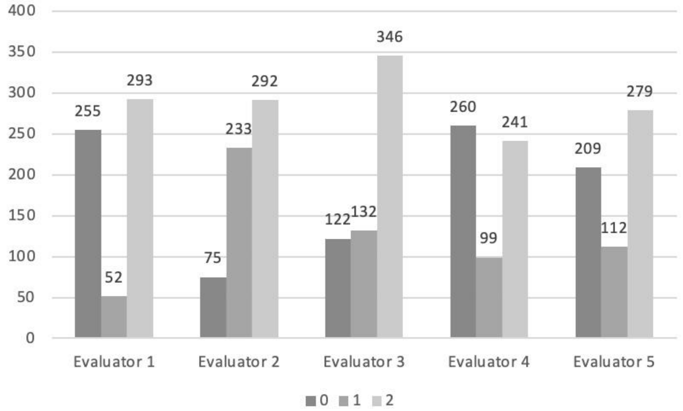


Fig. 3. The evaluation result.

TABLE 4
THE SPEARMAN'S CORRELATION COEFFICIENT FOR EVALUATORS

Evaluator	Evaluator 1	Evaluator 2	Ecaluator 3	Evaluator 4	Evaluator 5
Evaluator 1	1.00000	0.51427	0.43015	0.48874	0.49104
Evaluator 2	0.51427	1.00000	0.48255	0.56989	0.58914
Evaluator 3	0.43015	0.48255	1.00000	0.40326	0.48922
Evaluator 4	0.48874	0.56989	0.40326	1.00000	0.59525
Evaluator 5	0.49104	0.58914	0.48922	0.59525	1.00000

Examples of results that have semantic similarities are shown in Table 5. In dataset number 1, the word ”merupakan” is paraphrased as ”adalah”. The following are the candidate replacement words for ”merupakan”: 'adalah', 0.83, 'ialah', 0.57, 'bukanlah', 0.56, 'meliputi', 0.53, 'menjadi', 0.51, 'memiliki', 0.50, 'berasal', 0.50, 'mencakup', 0.49, 'hanyalah', 0.49, 'terdiri', 0.48, 'mempunyai', 0.48, 'menjadikannya', 0.46, 'mencerminkan', 0.45, 'yaitu', 0.44, 'yakni', 0.43. The word "adalah" is the word with the highest similarity score and is not in the antonym list.

TABLE 5
EXAMPLES OF RESULTS THAT HAVE SEMANTIC SIMILARITIES

No	Input	Output
1	*Pancasila merupakan dasar negara kita*	*Pancasila adalah dasar negara kita*
2	*Kami tinggal di Jakarta*	*Kami menetap di Jakarta*
3	*Ibu pergi*	*Ibu berangkat*
4	*Pameran itu akan dibuka oleh Ibu Gubernur*	*Pameran itu akan diresmikan oleh Ibu Gubernur*

In dataset number 2, the word ”tinggal” is paraphrased as ”menetap”. The following are candidate replacement words for "tinggal": 'menetap', 0.83, 'bermukim', 0.75, 'singgah', 0.66, 'berdiam', 0.65, 'berdomisili', 0.64, 'menginap', 0.63, 'tinggalnya', 0.63, 'berlindung', 0.61, 'bermalam', 0.60, 'bersembunyi', 0.59, 'berlibur', 0.58, 'dibesarkan', 0.58, 'bersekolah', 0.57, 'berkeliaran', 0.57, 'menghuni', 0.56. The word "menetap" is the word with the highest similarity score, and is not in the antonym list.

In dataset number 3, ”pergi” is paraphrased as “berangkat”. The following are candidate replacement words for “pergi”: 'bergegas', 0.77, 'pulang', 0.77, 'berangkat', 0.76, 'mengirimnya', 0.75, 'menemaninya', 0.73, 'datang', 0.71, 'membawanya', 0.70, 'menyelinap', 0.70, ‘mengikutinya', 0.70, 'bepergian', 0.69, 'mampir', 0.69, 'membuangnya', 0.68, 'mengembara', 0.66, 'mengantarnya', 0.66, 'diantar', 0.65. The words “bergegas” and “pulang” are words that have a high similarity score, but both are included in the antonym list so they cannot replace the word “pergi”. Therefore, the substitute word chosen is the word “berangkat”.

In dataset number 4, ”dibuka” is paraphrased as “diresmikan”. The following are candidate replacement words for “dibuka”: 'ditutup', 0.74, 'diresmikan', 0.70, 'dibangun', 0.68, 'direnovasi', 0.67, 'dibukanya', 0.63, 'diluncurkan', 0.60, 'didirikan', 0.60, 'diadakan', 0.58, 'dibukalah', 0.56, 'dioperasikan', 0.56, 'dibongkar', 0.56, 'dimulai', 0.56, 'beroperasi', 0.56, 'membuka', 0.55, 'dilaksanakan', 0.55. The words “dibuka” and “ditutup” are surprisingly similar in language models, even though they have opposite meanings semantically. However, because both are listed in the antonym list, the word chosen is the word “diresmikan”. If the word “dibuka” is contained in the word "Pintu dibuka saya", then of course the result of the paraphrase "Pintu diresmikan saya" has a different meaning.

Examples of results that have semantic similarities are shown in Table 6. In dataset number 1 the word ”berenang” is paraphrased as ”menyelam”. The following are candidate replacement words for “berenang”: 'menyelam', 0.82, 'mendayung', 0.70, 'berolahraga', 0.66, 'memancing', 0.65, 'berendam', 0.65, 'mengapung', 0.65, 'berselancar', 0.64, 'berjemur', 0.63, 'merangkak', 0.62, 'snorkeling', 0.62, 'bernapas', 0.62, 'berlari', 0.61, 'melompat', 0.60, 'bersantai', 0.60, 'merayap', 0.59. The word “berenang” does not have exactly the same synonyms. According to the KBBI, the word “mengapung” or “menyelam” has close meaning semantically.

TABLE 6
EXAMPLES OF RESULTS THAT HAVE PARTIAL SEMANTIC SIMILARITIES

No	Input	Output
1	*Kami berenang*	*Kami menyelam*
2	*Tiga ratus tiga belas tentara Islam mengalahkan ribuan tentara Quraisy*	*Tiga ratus tiga belas tentara Islam mengungguli ribuan tentara Quraisy*
3	*Dia telah merugikan kita satu setengah hari*	*Dia telah membahayakan kita satu setengah hari*
4	*Saya mengambil bunga*	*Saya mendapatkan bunga*

In dataset number 2 the word “mengalahkan” is paraphrased as “mengungguli”. The following are candidate replacement words for ”mengalahkan”: 'dikalahkan', 0.80, 'mengalahkannya', 0.72, 'mengungguli', 0.68, 'menundukkan', 0.66, ‘menaklukkan', 0.61, 'kalah', 0.61, 'mengandaskan', 0.60, 'melawan', 0.60, 'menang', 0.60, 'menaklukan', 0.60, 'kalahkan', 0.59, 'menyisihkan', 0.57, 'berduel', 0.57, 'bertarung', 0.57, 'menyingkirkan', 0.57. The word “dikalahkan” and “mengalahkannya” were not selected by the system because both have the same root word as the word “mengalahkan”. The word “menundukkan” or “menaklukkan” are closer in meaning semantically, but their similarity score are lower than the word “mengungguli”.

In dataset number 3 the word “merugikan” is paraphrased as “membahayakan”. The following are candidate replacement words for “merugikan”: 'membahayakan', 0.77, 'menguntungkan', 0.73, 'membebani', 0.68, 'mengganggu', 0.67, 'mengkhawatirkan', 0.66, 'memberatkan', 0.66, 'mempersulit', 0.65, 'diuntungkan', 0.65, 'dirugikan', 0.64, 'mempengaruhi', 0.63, 'memengaruhi', 0.63, 'memperparah', 0.63, 'memperburuk', 0.62, 'menyulitkan', 0.62,'merusak', 0.62. In this example, the word “merugikan” does not have an exact equivalent word. The system chooses the word “membahayakan” according to the highest similarity score.

In dataset number 4 the word “mengambil” is paraphrased as “mendapatkan”. The following are candidate replacement words for “mengambil”: 'diambil', 0.65, 'ambil', 0.63, ‘diambilnya', 0.59, 'mendapatkan', 0.59, 'memberinya', 0.58, 'menerima', 0.56, ‘memberi', 0.56, 'megambil', 0.55, ‘memperoleh', 0.54, 'membalikkan', 0.53, 'mendapat', 0.53, ‘meletakkan', 0.52, 'mengembalikan', 0.52, 'memegang', 0.52, 'mencuri', 0.51. Based on the context, the word “mengambil” can have many equivalents, and the word "mendapatkan" should be the best substitute in the context of the sentence.

Examples of results that have semantic similarities are shown in Table 7. In dataset number 1, the word ‘’mengosongkan” is paraphrased as “memindahkan”. The following are candidate replacement words for ‘mengosongkan’: 'memindahkan', 0.67, 'membersihkan', 0.66, 'membuang', 0.64, 'mengubur', 0.63, 'menyegel', 0.61, 'mengotori', 0.61, 'membakar', 0.60, 'melepaskan', 0.60, 'melepaskannya', 0.59, ‘mengamankan', 0.59, ‘melepas', 0.59, 'meratakan', 0.59, 'merelokasi', 0.59, 'membentengi', 0.58, 'menyerahkan', 0.58. There is no right word to replace the word “mengosongkan” in the context of the sentence. The system had chosen the best word it could come up with.

TABLE 7
Examples of results that have no semantic similarities

No	Input	Output
1	*Anda mengosongkan pundi-pundi*	*Anda memindahkan pundi-pundi*
2	*Jalur-jalur logistik dibangun dari satu wilayah ke wilayah lain*	*Jalur-jalur logistik direnovasi dari satu wilayah ke wilayah lain*
3	*Nenek tak perlu melakukan apa-apa*	*Nenek tak perlu mengadakan apa-apa*
4	*Ia duduk dengan angkuhnya*	*Ia berbaring dengan angkuhnya*

In dataset number 2, the word ”dibangun” is paraphrased as “direnovasi”. The following are candidate replacement words for ‘dibangun’: 'direnovasi', 0.74, 'dipugar', 0.69, 'dibuka', 0.68, 'dibangunnya', 0.66, 'direstorasi', 0.65, 'dibangunlah', 0.63, 'didirikan', 0.63, 'diresmikan', 0.62, 'dibongkar', 0.66, 'diruntuhkan', 0.61, 'dirancang', 0.61, 'ditempati', 0.60, 'dikembangkan', 0.60, 'dibagun', 0.60, 'membangun', 0.60. The correct word to replace the word "dibangun" is “didirikan”.

In dataset number 3, the word ”melakukan” is paraphrased as “mengadakan”. The following are candidate replacement words for ‘melakukan’: 'mengadakan', 0.74, 'dilakukan', 0.71, 'melalukan', 0.71, 'dilakukannya', 0.70, 'melaksanakan', 0.68, 'melancarkan', 0.65, 'menghentikan', 0.64, 'lakukan', 0.63, 'merencanakan', 0.62, 'menunda', 0.59, 'menjalankan', 0.58, 'menjalani', 0.57, 'menggelar', 0.56, 'melangsungkan', 0.56, 'melanjutkan', 0.55. The correct word to replace the word "melakukan" is “melaksanakan”.

In dataset number 4, the word ”duduk” is paraphrased as “berbaring”. The following are candidate replacement words for ‘duduk’: 'berbaring', 0.69, 'tidur', 0.60, 'duduknya', 0.61, 'berlutut', 0.57, 'didudukkan', 0.55, 'berjongkok', 0.55, 'bersila', 0.54, 'diletakkan', 0.54, 'bangku', 0.53, 'tidurnya', 0.53, 'jongkok', 0.53, 'terbaring', 0.53, 'ditaruh', 0.52, 'berteduh', 0.52, 'meletakkannya', 0.51. There is no right word to replace the word “duduk” in the context of the sentence. The system had chosen the best word it could come up with.

There is a weakness in the system which causes it to be unable to generate paraphrased sentences from the sentences inputed, namely Pos Tagger cannot always recognize VB in the PoS tagging process. The sentences “Saya ditugasi pekerjaan itu oleh dia” are tagged as follows: ('Saya', 'PRP'), ('ditugasi', 'NN'), ('pekerjaan', 'NN'), ('itu', 'PR'), ('oleh', 'IN'), ('dia', 'PRP'), so that the input sentence cannot be paraphrased. If 'ditugasi' is identifiable as VB, then the system paraphrases it as “Saya diperintahkan pekerjaan itu oleh dia”. PoS tagger also cannot recognize VB in sentences where the verb contains a possessive pronoun, as in the example: ‘Anaknya sedang diajarnya’. The postagging results for the sentence are ('Anaknya', 'RB'), ('sedang', 'MD'), ('diajarnya', 'RB'). The word “diajarnya” tagged as RB (possessive pronoun) (Table 1).

Is there always a paraphrase for every sentence input? In the following example sentence “orang itu sedang tidur”, the word 'tidur' is identified as VB. Then the system would paraphrase it as ‘orang itu sedang berbaring’. Is there a more semantic equivalent to the word 'tidur'? In KBBI the correct equivalent for 'tidur' is in fact 'berbaring' or ‘terbaring (tidak berdiri)'. But in reality, the meaning of ‘berbaring’ cannot completely replace the meaning of 'tidur'. ‘Tidur’ implies resting the body and consciousness usually by closing the eyes.

IV. Conclusion

In this research a system has been built that can generate paraphrases for simple sentences in Indonesian. Using 600 datasets, the five evaluators show that 48.37% of the sentences have semantic similarities, 20.93% have semantic reductions, and 30.70% have no semantic similarities. The result is due to the quality of the PoS Tagger, the completeness vocabulary of model languages, and the completeness of antonym word list. The PoS Tagger used is recommended to use the latest PoS Tagger. The completeness of the vocabulary in the language model is due to the corpus used being the corpus from Wikipedia. As time goes by, the Indonesian Wikipedia corpus will increase in amount of vocabulary.

References

[1] Badan Pengembangan dan Pembinaan Bahasa, Kementerian Pendidikan, Kebudayaan, Riset dan Teknologi Republik Indonesia, *Kamus Besar Bahasa Indonesia KBBI*, [online] Available at: http:// kbbi.web.id [Accessed 10 October 2022], 2016.

[2] R. Bhagat and E. Hovy, “What is a Paraphrase?” in *Computational Linguistics*, 2013, 39, 3, pp. 463-472.

[3] G. Hintz, “Data-driven Paraphrasing and Stylistic Harmonization” in *Proceedings of NAACL-HLT*, San Diego, California: *Association for Computational Linguistics*, 2016, pp. 37-44.

[4] Xu, W., Ritter, A., Dollan, W. B., Grishman, R., and Cherry, C. Paraphrasing for Style. *Proceedings of COLING 2012: Technical Papers, pp. 2899–2914*. Mumbai: ACL. 2012.

[5] Kaji, N., Okamoto, M., and Kurohashi, S. “Paraphrasing Predicates from Written Language to Spoken Language Using the Web”. *Human Language Technology Conference of the North American Chapter HLT NAACL* pp. 241-248. Boston: the Association for Computational Linguistics. 2004

[6] L. Dong, J. Mallinson, S. Reddy, and M. Lapata. 2017. Learning to paraphrase for question answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 875–886.

[7] S. Zhu, X. Cheng, S. Su, and Sh. Lang. 2017. Knowledge-based question answering by jointly generating, copying and paraphrasing. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 2439–2442.

[8] R. M. Seraj, M. Siahbani, and A. Sarkar. 2015. Improving statistical machine translation with a multilingual paraphrase database. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1379–1390.

[9] B. Thompson and M. Post. 2020a. Automatic machine translation evaluation in many languages via zero-shot paraphrasing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP),* pp. 90–121.

[10] A. A. Suryani, D. H. Widyantoro, A. Purwarianti, and Y. Sudaryat. (2015). Experiment on a phrase-based statistical machine translation using PoS Tag information for Sundanese into Indonesian. *International Conference on Information Technology Systems and Innovation (ICITSI)* (pages 1-6). Bandung: IEEE.

[11] Y. Cao and X. Wan. 2020. Divgan: Towards diverse paraphrase generation via diversified generative adversarial network. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pp. 2411–2421.

[12] A. Kumar, S. Bhattamishra, M. Bhandari, and P. Talukdar. 2019. Submodular optimization-based diverse paraphrasing and its effectiveness in data augmentation. In *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), pp. 3609–3619.

[13] S. Gao, Y. Zhang, Zh. Ou, and Zh. Yu. 2020. Paraphrase augmented task-oriented dialog generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 639–649.

[14] Barmawi, A. M., and Muhammad, A. Paraphrasing Method Based on Contextual Synonym Substitution. *J. ICT Res. Appl.*, 257-282. 2019.

[15] Y. Wibisono, “POS Tagger Bahasa Indonesia dengan Python”. [Online]. Available: https://yudiwbs.wordpress.com/2018/02/20/pos-tagger-bahasa-indonesia-dengan-pytho/ [accessed: 4 Oct 2021].

[16] Bunyamin, A. F. Huda, and A. A. Suryani. Indonesian Stemmer for ambiguous word based on context. *ICODSA 2021*. p. 1-9.