Xi'an ShiyouDaxueXuebao (ZiranKexue Ban)/
Journal of Xi'an Shiyou University, Natural Sciences Edition
ISSN:1673-064X
E-Publication: Online Open Access
DOI 10.17605/OSF.IO/BHD3J

NON-STANDARD WORDS DETECTION SYSTEM IN A TEXT WITH WORD MATCHING METHOD

DES SURYANI[1] [2], AMBIYAR[1], ASRUL HUDA[1], WILDA SRIHASTUTY HANDAYANI PILIANG[3], RIKA MELYANTI[1] [4], FITRI AYU[1] [5]
[1]Doctoral Program of Technology and Vocational Education, Faculty of Engineering - Universitas Negeri Padang, Indonesia
[2]Study Program of Informatics Engineering, Faculty of Engineering - Islamic University of Riau, Indonesia
[3]Study Program of Indonesian Language, Faculty of Education - Islamic University of Riau, Indonesia
[4]Study Program of Information System, STMIK Hang Tuah Pekanbaru, Indonesia
[5]Study Program of Information Management, AMIK Mahaputra Riau, Indonesia
Email: des.suryani@eng.uir.ac.id

Abstract
Indonesian is the state language which is the official language of the Unitary State of the Republic of Indonesia. The Indonesian language used must be standard or standard Indonesian and by good and correct Indonesian spelling rules. In its implementation in the field, there are still many errors in the standard language in the development of the national culture, science, and technology. An example is found in the use of the wrong standard words in writing scientific essays in Indonesian. This happens because of the lack of mastery of standard vocabulary among writers. In addition, it can also occur due to the habit of people who often pick up the language around them without any filtering process first. The languages that are commonly used are considered correct and are never interested in finding out the origin or meaning of the language. In the end, what happened was the use of the wrong Indonesian language was used for generations. Based on the phenomena that occur in society, it is necessary to research to build a non-standard word shortening system using the Python programming language and the Approximate String Matching method. This system will match the words in the non-standard word bag of words (TB) with the abstract text. Abstract data used as samples are abstracts from texts (reports, papers, scientific works) provided that the number of words in the abstract does not exceed 200 words. The results of this study can find and determine the number of non-standard words contained in one or several abstracts and replace them with standard words.

Keywords— standard words, non-standard words, bag-of-words, approximate string matching

1 Introduction

The Indonesian nation is a nation that is rich in various languages. For that, we need a language that functions as a unifying language [1]. Indonesian is the state language which is the official language of the Unitary State of the Republic of Indonesia. This is stated in the 1945 Constitution chapter XV article 36 which reads "The State Language is Indonesian". The implications of Article 36 of the 1945 Constitution can be seen in that one of the functions of the Indonesian language in its position as the state language is a means of developing national culture, science, and technology. The Indonesian language used must be standard or standard Indonesian and by good and correct Indonesian spelling rules. This is because the function of Indonesian as the state language is used in a variety of formal or official situations, both spoken and written [2].

ISSN:1673-064X
E-Publication: Online Open Access
DOI 10.17605/OSF.IO/BHD3J

Standard Indonesian or in the form of standard words means words that are appropriate and used in the guidelines or rules of the Indonesian language that have been determined. Good and correct language can be interpreted as the use of a variety of languages that are in line with the target and in addition to following the correct language rules. Standard words are also explained as those whose spelling and rules are by the correct Indonesian language rules. We can see the source of the standard rules in the KBBI or Big Indonesian Dictionary Edition V (latest) either online, offline, or printed in book form. If the word in question is not listed in the KBBI, it can be ascertained that the word is not standard [3].

In its implementation in the field, there are still many errors in the standard language in the development of the national culture, science, and technology. An example is found in the use of the wrong standard words in writing scientific essays in Indonesian. This happens because of the lack of mastery of standard vocabulary among writers. In addition, it can also occur due to the habit of people who often pick up the language around them without any filtering process first. The languages that are commonly used are considered correct and are never interested in finding out the origin or meaning of the language. In the end, what happened was the use of the wrong Indonesian language was used for generations.

String matching is broadly divided into two (2), namely 1) matching the string precisely with the arrangement of characters in the string being matched; and 2) matching the strings vaguely, namely string matching where the matched strings have similarities but both have the same character arrangement. Different from [4] explains Approximate string matching can be used to search for strings based on the same string and strings that have similar writing to the strings in the dictionary. This method can be used for non-standard word search because it can identify the same string and have similar writing. [5] add one of the algorithms that can be used to search strings in approximate string matching is Levenshtein Distance. Levenshtein distance is an algorithm that can be used to detect similarities between two strings that have the potential to commit plagiarism.

Based on the phenomena that occur in society, the authors are interested in building a non-standard word shortening system with the word matching method which was compiled in a study entitled "Unstandard Word Detection System in a Text With Word Matching Method". The writer considers this research necessary because of the condition of the people who tend to ignore the importance of using standard Indonesian in research for the development of science and technology.

2 Research Methods

2.1 Data Collection

Abstract data used as samples are abstracts from texts (reports, papers, scientific works) provided that the number of words in the abstract should not exceed 200 words. Data was collected using descriptive methods and content analysis, namely by collecting and sorting out words, phrases, clauses, and sentences that were not by the Indonesian spelling rules in the abstract list which were then modeled in non-

Xi'an ShiyouDaxueXuebao (ZiranKexue Ban)/
Journal of Xi'an Shiyou University, Natural Sciences Edition
ISSN:1673-064X
E-Publication: Online Open Access
DOI 10.17605/OSF.IO/BHD3J

standard words bag-of-words [6]. Then, the wrong words, phrases, clauses, and sentences are analyzed according to their form. After that, the words, phrases, clauses, and sentences that have been analyzed are corrected and replaced with words, phrases, clauses, and sentences that should be (correct) appropriate and used in the guidelines or rules of the correct Indonesian language. The source of the standard rules can be seen in the KBBI or the Big Indonesian Dictionary Edition V (latest) either online, offline, or printed in book form. If the word in question is not listed in the KBBI, it can be ascertained that the word is not standard. The standard word list is then modeled in the standard word bag of words.

2.2 Data analysis method

To analyze the data, the researcher will build a non-standard word detection system with the Python Programming Language and use the Approximate String Matching Method [7]. The data that has been obtained must go through the initial stage of data processing/pre-processing before it can be analyzed as shown in Fig 1 below.

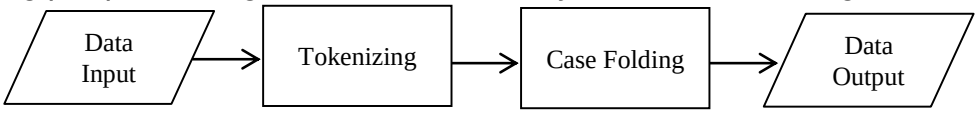


Fig 1. Stages *Pre-Processing*

Based on Fig 1, it can be explained that the initial processing stages in this study are as follows: 1) Input data, in the form of abstract data that will be tested for non-standard words; 2) Tokenizing, in the text data each word will be separated based on the space found. The tokenizing stage is the stage of cutting the input string based on each word that composes it. An example of this stage can be seen in Fig 2; 3) Case folding, the text data is changed from uppercase letters to lowercase letters and removes all punctuation marks in sentences. Not all text documents are consistent in the use of capital letters. Therefore, the role of Case Folding is needed in converting the entire text in the document into a standard form (usually lowercase or lowercase). For example, a user who wants to get "COMPUTER" information and types "KOMPUTER" or "computer", is still given the same retrieval result, namely "computer". Case folding is changing all the letters in the document to lowercase. Only letters 'a' to 'z' are accepted. Characters other than letters are omitted and are considered delimiters.

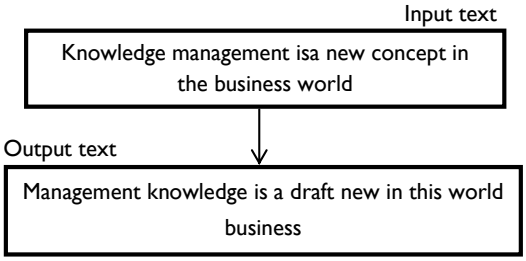


Fig 2. Stages Tokenizing

Xi'an ShiyouDaxueXuebao (ZiranKexue Ban)/
Journal of Xi'an Shiyou University, Natural Sciences Edition
ISSN:1673-064X
E-Publication: Online Open Access
DOI 10.17605/OSF.IO/BHD3J

Tokenization in Fig 2 outlines a set of characters in a text into word units, how to distinguish certain characters that can be treated as word separators or not. For example whitespace characters, such as enter, tabulation, space is considered word separators. However, for single quotes (‘), period (.), semicolon (;), colon (:) or others, it can have quite a lot of roles as word separators; and 4) Output data, in the form of abstract text data that can be processed. After the abstract text has gone through the initial data processing, the approximate string matching method will be used by matching the words in the non-standard word bag-of-words (TB) with the abstract text. If it is found that the word X has a small edit distance value with the word TB according to the specified threshold value, then the word X may be a non-standard word. Then the approximate string matching method will be used again by matching the words in the standard word bag-of-words (B) with the X word. If the edit distance value for the standard word B and the word X is less than the edit distance value for the non-standard word TB and the word X, then the word X will be exchanged for the standard word B.

2.3 Levenshtein distance algorithm

The Levenshtein distance algorithm was discovered by a Russian scientist named Vladimir Levenshtein in 1963, this algorithm is also called the edit distance algorithm [8]. The edit distance calculation is obtained from the matrix used to calculate the number of string differences between two strings, as an example of the results of using this algorithm, the strings "computer" and "computer" have a distance of 1 because only one operation is needed to change one string to another string. In the case of the two strings above, the string "computer" can become "computer" by simply changing the character "c" to "k". A Dimensional two (2) matrix is used in calculating the Levenshtein Distance value. The value in the matrix is the number of deletion, insertion, and exchange operations required to convert the source string to the target string. The operation formula for inserting, deleting, and exchanging characters used to fill in the matrix values is as follows [9-12].

D (s,t) = min D(s-1,t) + 1 (Deletion) (1)
D (s,t) = min D(s,t-1) + 1 (Insertion) (2)
D (s,t) = min D(s-1,t-1) + 1, sjti (Exchange) (3)
D (s,t) = min D(s-1,t-1), sj=ti (No change) (4)[1]

s = Source strings
(j) = Source string character to – j
t = Target strings
(i) = Target string character to – i
D = Levenshtein Distance

[1] Source: https://web.stanford.edu/class/cs124/lec/med.pdf

Xi'an ShiyouDaxueXuebao (ZiranKexue Ban)/
Journal of Xi'an Shiyou University, Natural Sciences Edition
ISSN:1673-064X
E-Publication: Online Open Access
DOI 10.17605/OSF.IO/BHD3J

3 Results and Discussion

In this section are the steps of implementation and discussion of the results of the research conducted. Implementation plans need to be made in advance so that implementation goes well and by the expected goals. This implementation plan intends to regulate the detection of non-standard words into standard words in abstract data.

3.1 Abstract data

Abstract 1:
Spelling errors in text editors are common. Spelling error checks are usually done when the writing has been completed. In a short text, it is certainly not difficult. However, when the text size has reached more than ten thousand words or even millions of words, checking this way is very difficult. With the implementation of a program, these problems can be overcome. In a small scope, student activities, this application will be very useful, for example in the correction of written works, theses, and so on. An efficient algorithm for correcting errors by dynamic programming based on string comparison will be discussed further in this paper.

Abstract 2:
SMA Negeri 1 Padang conducted a psychological test using the Intelligence Structure Test (IST) as a psychological test tool to determine the classification of students' IQ. The IQ classification of these students is served by a psychologist. The results of this student's IQ classification have not been evaluated to obtain knowledge from the data set. For this reason, it is necessary to research to see future opportunities to help psychologists classify students' IQ. In this case, applying the concept of data mining with the Naive Bayes method and using the David Weschler scale as a research instrument. Based on testing 100 test data on 232 and 332 test data, this study was able to classify the participants' IQ with an accuracy rate of more than 90%. With this system, IQ classification can be done more quickly and accurately.

Both the abstracts above are stored in a file with the name Abstract.csv. To read the data can be used with the instructions:

```
posts = pd.read_csv(open('Abstract.csv', newline='', encoding='utf-8'), delimiter=';')
```

The contents of the abstract.csv file can be done with the statement:

```
print (posts.shape)
posts head(100)
```

And the results of the display of the contents of the file can be seen in Fig 4 below.

	No	Abstrak
0	1	Kesalahan pengejaan pada editor teks sering k...
1	2	SMA Negeri 1 Padang menyelenggarakan tes psiko...

Fig 4. Abstract data content.csv (In Indonesia).

Xi'an ShiyouDaxueXuebao (ZiranKexue Ban)/
Journal of Xi'an Shiyou University, Natural Sciences Edition
ISSN:1673-064X
E-Publication: Online Open Access
DOI 10.17605/OSF.IO/BHD3J

3.2 Preprocessing

In the next stage, preprocessing the abstract.csv file using tokenizing and lower case. Tokenizing is used to separate word by word into tokens, the next is lower case, which is changing capital letters to lowercase letters. Then merged back into an abstract that is stored in a new file, in this case, named abstract_pure. The coding of the program is as follows:

```
def identify_tokens(row):
   review = row['Abstrak']
   tokens = nltk.word_tokenize(review.lower())
   # taken only words (not punctuation)
   token_words = [w for w in tokens]
   return token_words
   posts['Words_Abstrak'] = posts.apply(identify_tokens, axis=1)

# Join abstrak
def rejoin_words(row):
   my_list = row['Words_Abstrak']
    joined_words = ( " ".join(my_list))
   return joined_words
   posts['Abstrak_Murni'] = posts.apply(rejoin_words, axis=1)
```

	No	Abstrak	Words_Abstrak	Abstrak_Murni
0	1	Kesalahan pengejaan pada editor teks sering k...	[kesalahan, pengejaan, pada, editor, teks, ser...	kesalahan pengejaan pada editor teks sering ka...

Fig 5. Preprocessing results (In Indonesia).

Fig 5 is the result of abstract preprocessing. The abstract column is abstract content that has not gone through preprocessing. The Words_Abstract column is the result of the tokenizing and lower case. While the Abstract_Murni column is the result of joining words into sentence forms like the previous abstract and all words are in lowercase.

3.3 Standard words and non-standard words

Standard and non-standard words are dictionaries used to detect non-standard words against abstract files that have gone through the preprocessing stage. The standard and non-standard word data dictionary are in the form of a standard_word_not_word.csv file for easy processing. The file can be opened and read with the following statement:

Xi'an ShiyouDaxueXuebao (ZiranKexue Ban)/
Journal of Xi'an Shiyou University, Natural Sciences Edition
ISSN:1673-064X
E-Publication: Online Open Access
DOI 10.17605/OSF.IO/BHD3J

```
dictionary =csv.reader(open('standard_word_not_ standard.csv', newline='', encoding='utf-8'), delimiter=';')
  ArrayDictionary=[]
  for row in dictionary:
  low =(row[0].lower(), row[1].lower())
  word = tuple(low)
  ArrayDictionary.append(word)
```

To display all the data can be done with the following instructions:

```
View_dictionary = pd.Data Frame (Array Dictionary)
View_dictionary.head (10)
```

The results of the coding can be seen in Fig 6 below.

	0	1
0	baku	tidak_baku
1	abjad	abjat
2	advokat	adpokat
3	aktif	aktip
4	al quran	alquran
5	apotek	apotik
6	asas	azas
7	atlet	atlit
8	atmosfer	atmosfir
9	baut	baud

Figu 6. Fill in the standard – non-standard word dictionary (In Indonesia).

3.4 Iterative Levenshtein

In the next stage, create an Iterative Levenshtein function that is used as a comparison matrix for non-standard words and standard words contained in the abstract. This function will display the value of the number of errors in writing words from the matrix. The iterative_levenshtein function contains the following instructions:

```
defiterative_levenshtein(s, t):
rows = len(s)+1
cols = len(t)+1
dist = [[0 for x in range(cols)] for x in range(rows)]
      # source prefixes can be transformed into empty strings
      # by deletions:
for i in range(1, rows):
dist[i][0] = i
      # target prefixes can be created from an empty source string
      # by inserting the characters
for i in range(1, cols):
dist[0][i] = i
for col in range(1, cols):
```

ISSN:1673-064X
E-Publication: Online Open Access
DOI 10.17605/OSF.IO/BHD3J

```
for row in range(1, rows):
if s[row-1] == t[col-1]:
cost = 0
else:
cost = 1
dist[row][col] = min(dist[row-1][col] + 1, # deletion
dist[row][col-1] + 1, # insertion
dist[row-1][col-1] + cost) # substitution
arrayindex=[]
arraycolom=[]
arrayA=[]
   #display non-standard words
   d='0'
   y=d+word[1]
for later in y:
arrayindex.append(later)
   #display standard words
   [arraycolom.append(later) for later in d+word[0]]
   #show results
   for r in range(rows):
   arrayA.append(dist[r])
   xl=pd.DataFrame(data=arrayA,index=arrayindex,columns=arraycolom)
   print(xl)
   returndist[row][col]
```

3.5 Abstract repair

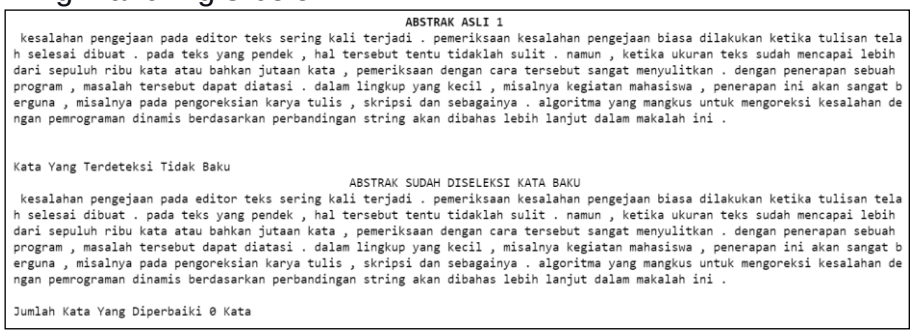
The next step is to check the words in an abstract. When non-standard words are found, they will be corrected automatically and the number of errors in writing non-standard words will be calculated and the comparison in writing non-standard words with standard words in a matrix using the Levenshtein Iterative Function.

```
# Matrix
plus=0
for fix in posts['Pure_Abstract']:
plus+=1
print("\033[1m"+'\t\t\t\t\t\t\tORIGINAL ABSTRACT'+"\033[0;0m",plus,'\n',fix,'\n',
'\n\nWords Detected Not
Default')
count=0
for words in ArrayDictionary:
   #word[1] is a non-standard word, word[0] is a standard word
if word[1] in fix:
count+=1
matrix = iterative_levenshtein(word[1],word[0])
repair = fix.replace(word[1],"\033[1;41m"+word[0]+"\033[1;m")
```

Xi'an ShiyouDaxueXuebao (ZiranKexue Ban)/
Journal of Xi'an Shiyou University, Natural Sciences Edition
ISSN:1673-064X
E-Publication: Online Open Access
DOI 10.17605/OSF.IO/BHD3J

```
print("\nThe value does not match",word[1],":",matrix,"\n")
print('\t\t\t\t\t\tABSTRACT ALREADY SELECTED STANDARD
WORDS\n',correct,'\n\nNumber of Cor
rected Words',count,'Words\n\n')
```

Based on the program code, the results of non-standard word detection can be obtained as in Fig 7 and Fig 8 below.



ABSTRAK ASLI 1

kesalahan pengejaan pada editor teks sering kali terjadi . pemeriksaan kesalahan pengejaan biasa dilakukan ketika tulisan telah selesai dibuat . pada teks yang pendek , hal tersebut tentu tidaklah sulit . namun , ketika ukuran teks sudah mencapai lebih dari sepuluh ribu kata atau bahkan jutaan kata , pemeriksaan dengan cara tersebut sangat menyulitkan . dengan penerapan sebuah program , masalah tersebut dapat diatasi . dalam lingkup yang kecil , misalnya kegiatan mahasiswa , penerapan ini akan sangat berguna , misalnya pada pengoreksian karya tulis , skripsi dan sebagainya . algoritma yang mangkus untuk mengoreksi kesalahan dengan pemrograman dinamis berdasarkan perbandingan string akan dibahas lebih lanjut dalam makalah ini .

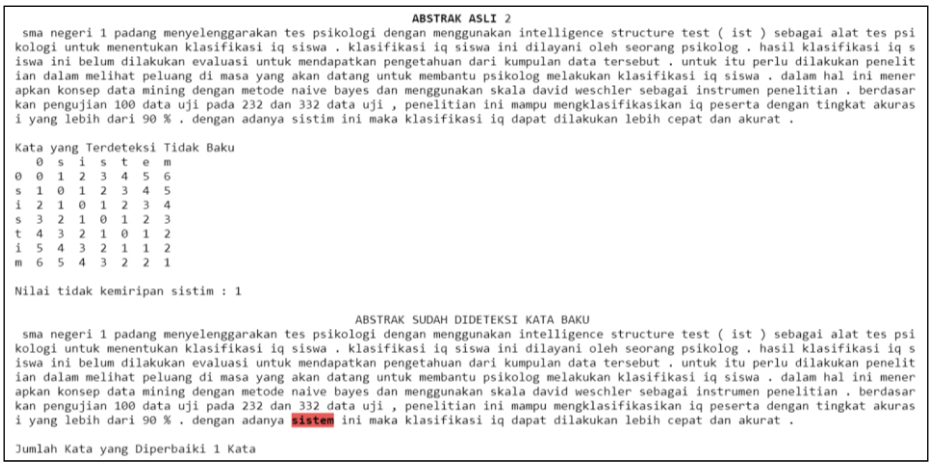
Kata Yang Terdeteksi Tidak Baku

ABSTRAK SUDAH DISELEKSI KATA BAKU

kesalahan pengejaan pada editor teks sering kali terjadi . pemeriksaan kesalahan pengejaan biasa dilakukan ketika tulisan telah selesai dibuat . pada teks yang pendek , hal tersebut tentu tidaklah sulit . namun , ketika ukuran teks sudah mencapai lebih dari sepuluh ribu kata atau bahkan jutaan kata , pemeriksaan dengan cara tersebut sangat menyulitkan . dengan penerapan sebuah program , masalah tersebut dapat diatasi . dalam lingkup yang kecil , misalnya kegiatan mahasiswa , penerapan ini akan sangat berguna , misalnya pada pengoreksian karya tulis , skripsi dan sebagainya . algoritma yang mangkus untuk mengoreksi kesalahan dengan pemrograman dinamis berdasarkan perbandingan string akan dibahas lebih lanjut dalam makalah ini .

Jumlah Kata Yang Diperbaiki 0 Kata

Fig 7. Non-standard word detection results without writing errors (In Indonesia).



ABSTRAK ASLI 2

sma negeri 1 padang menyelenggarakan tes psikologi dengan menggunakan intelligence structure test (ist) sebagai alat tes psikologi untuk menentukan klasifikasi iq siswa . klasifikasi iq siswa ini dilayani oleh seorang psikolog . hasil klasifikasi iq siswa ini belum dilakukan evaluasi untuk mendapatkan pengetahuan dari kumpulan data tersebut . untuk itu perlu dilakukan penelitian dalam melihat peluang di masa yang akan datang untuk membantu psikolog melakukan klasifikasi iq siswa . dalam hal ini menerapkan konsep data mining dengan metode naive bayes dan menggunakan skala david weschler sebagai instrumen penelitian . berdasarkan pengujian 100 data uji pada 232 dan 332 data uji , penelitian ini mampu mengklasifikasikan iq peserta dengan tingkat akurasi yang lebih dari 90 % . dengan adanya sistim ini maka klasifikasi iq dapat dilakukan lebih cepat dan akurat .

Kata yang Terdeteksi Tidak Baku

	0	s	i	s	t	e	m
0	0	1	2	3	4	5	6
s	1	0	1	2	3	4	5
i	2	1	0	1	2	3	4
s	3	2	1	0	1	2	3
t	4	3	2	1	0	1	2
i	5	4	3	2	1	1	2
m	6	5	4	3	2	2	1

Nilai tidak kemiripan sistim : 1

ABSTRAK SUDAH DIDETEKSI KATA BAKU

sma negeri 1 padang menyelenggarakan tes psikologi dengan menggunakan intelligence structure test (ist) sebagai alat tes psikologi untuk menentukan klasifikasi iq siswa . klasifikasi iq siswa ini dilayani oleh seorang psikolog . hasil klasifikasi iq siswa ini belum dilakukan evaluasi untuk mendapatkan pengetahuan dari kumpulan data tersebut . untuk itu perlu dilakukan penelitian dalam melihat peluang di masa yang akan datang untuk membantu psikolog melakukan klasifikasi iq siswa . dalam hal ini menerapkan konsep data mining dengan metode naive bayes dan menggunakan skala david weschler sebagai instrumen penelitian . berdasarkan pengujian 100 data uji pada 232 dan 332 data uji , penelitian ini mampu mengklasifikasikan iq peserta dengan tingkat akurasi yang lebih dari 90 % . dengan adanya sisten ini maka klasifikasi iq dapat dilakukan lebih cepat dan akurat .

Jumlah Kata yang Diperbaiki 1 Kata

Fig 8. Non-standard word detection results without writing errors (In Indonesia).

4 Conclusions

Based on the results of the analysis and discussion conducted on the standard word detection system in the abstract, the following conclusions can be drawn 1) Standard word detection can be done by combining the approximate string matching method and the Levenshtein distance algorithm; 2) The test results can produce the number of non-standard words contained in one or more texts being tested; and 3)This study uses a limited number of standard words.

ISSN:1673-064X
E-Publication: Online Open Access
DOI 10.17605/OSF.IO/BHD3J

5 Acknowledgment

This research is one of the publication requirements for my dissertation at the Doctoral Program of Technology and Vocational Education, Faculty of Engineering - Universitas Negeri Padang, Indonesia. The author would like to acknowledge the promoter and the head of the doctoral program who have given the author the opportunity to publish this initial research.

6 References

[1] Paauw, S. (2009). One land, one nation, one language: An analysis of Indonesia's national language policy. *University of Rochester Working Papers in the Language Sciences*, *5*(1), 2-16.

[2] Kang, Y. (2019). Statehood, Scale and Hierarchy: History, Language and Identity in Indonesia Lauren Zentz Bristol, UK: Multilingual Matters. 280 pp.

[3] Alwi, H., Dardjowidjojo, S., Lapoliwa, H., & Moeliono, A. (1993). Tata Bahasa Baku Bahasa Indonesia Edisi Kedua. *Jakarta: Departemen Pendidikan dan kebudayaan Republik Indonesia*.

[4] Sagita, V., & Prasetiyowati, M. I. (2013). Studi Perbandingan Implementasi Algoritma Boyer-Moore, Turbo Boyer-Moore, dan Tuned Boyer-Moore dalam Pencarian String. *Ultimatics: Jurnal Teknik Informatika*, *5*(1), 31-37.

[5] Su, Z., Ahn, B. R., Eom, K. Y., Kang, M. K., Kim, J. P., & Kim, M. K. (2008, June). Plagiarism detection using the Levenshtein distance and Smith-Waterman algorithm. In *2008 3rd International Conference on Innovative Computing Information and Control*, 569-569.

[6] Fasold, R. W., & Connor-Linton, J. (Eds.). (2014). *An introduction to language and linguistics*. Cambridge university press.

[7] Kostakos, P., Moilanen, M., Niemelä, A., & Oussalah, M. (2017). Catchem: A Browser Plugin for the Panama Papers using Approximate String Matching. *European Intelligence and Security Informatics Conference (EISIC)*, 139-142.

[8] Berthele, R. (2011). On abduction in receptive multilingualism. Evidence from cognate guessing tasks. *Applied Linguistics Review*, *2*(2011), 191-220.

[9] Rosmala, D., & Risyad, Z. M. (2017). Algoritma Levenshtein Distance dalam Aplikasi Pencarian isu di Kota Bandung pada Twitter. *MIND (Multimedia Artificial Intelligent Networking Database) Journal*, *2*(2), 1-12.

[10] Melyanti, R., Giatman, M., & Mayliza, R. (2021). Online Determination of Credit Score (PAK) Application Functional Teachers. *International Journal of Management and Humanities (IJMH)*, 5(7), 89-93.

ISSN:1673-064X
E-Publication: Online Open Access
DOI 10.17605/OSF.IO/BHD3J

[11] Melyanti, R (2019). Aplikasi Penilaian Kinerja Dosen Dalam Pelaksanaan Tri Dharma Perguruan Tinggi (Studi Kasus: STMIK Hang Tuah Pekanbaru). *Jurnal Ilmu Komputer*, 8(1), 99-106

[12] Ayu, F., Aryatiningsih, D. S., Ambiyar, F. R., & Febriani, A. Design and Application of Detection Damage Hardware to Your Computer with Certainty Factor. 5(10), 28-31.