IJCCS (Indonesian Journal of Computing and Cybernetics Systems)
Vol.16, No.4, October 2022, pp. 401~410
ISSN (print): 1978-1520, ISSN (online): 2460-7258
DOI: https://doi.org/10.22146/ijccs.76396

Automatic Essay Scoring Using Data Augmentation in Bahasa Indonesia

Nur Fadilah*[1], Sigit Priyanta[2]
[1]Master Program of Computer Science, FMIPA UGM, Yogyakarta, Indonesia
[2]Department of Computer Science and Electronics, FMIPA UGM, Yogyakarta, Indonesia
e-mail: *[1]**nurfadilahderman@mail.ugm.ac.id**, [2]seagatejogja@ugm.ac.id

Abstrak

Penilaian tes uraian merupakan salah satu penilaian untuk mengetahui kemampuan siswa secara mendalam. UKARA merupakan pengembangan automatic essay scoring yang menggabungkan NLP dan machine learning. Penelitian ini menggunakan dataset yang disediakan untuk UKARA challenge yang terdiri dari 2 jenis, dataset A dan B. Dataset yang disediakan masih sedikit untuk proses pembuatan model sehingga menjadi salah satu penyebab model yang dihasilkan belum maksimal.

Penelitian ini fokus pada proses penambahan atau augmentasi data dengan menggunakan EDA (Easy Data Augmentation Techniques). Ada empat metode yang diterapkan yaitu Synonym Replacement (SR), Random Insertion (RI), Random Swab (RS), dan Random Deletion (RD). Data yang digunakan untuk pembuatan model dengan menggunakan metode BiLSTM. Performa model dievaluasi menggunakan confusion matrix dengan nilai accuracy, precision, recall dan f-measure.

Hasil penelitian menunjukkan bahwa pada dataset A tanpa augmentasi dengan menggunakan k-fold cross validation mengasilkan nilai akurasi tertinggi dengan nilai 85.07% . Sedangkan hasil pada data B menunjukkan EDA insert dengan k-fold cross validation sebesar 72.78%,

Kata kunci— *Tes Uraian Singkat, Augmentasi Data, Fasttext, BiLSTM*

Abstract

Essay is one of the assessments to find out the abilities of students in depth. UKARA is an automatic essay scoring development that combines NLP and machine learning. This study uses the datasets provided for the UKARA challenge which consists of 2 types, datasets A and B. The dataset provided is still small for the model creation process so that it is one of the causes of the resulting model is not optimal.

This research focuses on the process of adding or augmenting data using EDA (Easy Data Augmentation Techniques). There are four methods applied, namely Synonym Replacement (SR), Random Insertion (RI), Random Swab (RS), and Random Deletion (RD). The data is used for model creation by using the BiLSTM method. Performa model evaluated using confusion matrix with nilai accyouracy, precision, recall dan f-measure.

The results showed that the dataset A without augmentation using k-fold cross validation produced the highest accuracy value with a value of 85.07%. While the results in data B show EDA insert with k-fold cross validation of 72.78%,

Keywords—*Test Brief Description, Data Augmentation, Fasttext, BiLSTM*

Received July 19th,2022; Revised July 25th, 2022; Accepted October 31th, 2022

1. INTRODUCTION

Education and assessment are two inseparable things. Improving the quality of education can be achieved by improving the quality of learning and the quality of the assessment system [1]. Essay is one of the assessments that are often used by educators to find out the abilities of students in depth. Essay is a form of subjective assessment used to test the learning ability of the nature of discussion and description of words [2]. The development of technology in the form of Artificial Intelligence at the moment makes it possible to create a essay test scoring system automatically, looking at some previously related research on the assessment of essay tests automatically by using the Generalized Latert Semantic Algorithm (GLSA) method [3] and another study is research using the Knowledge discovery in Text (KDT) method with a modication Porter Stemming Algorithm [4]. Both studies focused on the degree of text similarity between the answer key and the existing answer. So it has the disadvantage interpreting the meaning along with the content of the answer key with the actual answer, while the purpose of the essay scoring system is to find out a person's cognitive abilities in the form of understanding the material.

Further research raised the same theme as the name of the system is UKARA [5]. UKARA in its application combines NLP with supervised machine learning processes. The questions in the test are tested first independently without machine learning, then produce student answers, these student answers are then labeled true and false into a dataset manually by the assessment team. The dataset is then processed through machine learning so that the system is able to learn a variety of true and false answers. UKARA has been developed from 2018 till now. Development is carried out by applying several approaches ranging from the word embedding approach to detect vectorization of similarities of words and sentences in dataset texts and using deep learning methods in the development of the model.

From several developments carried out in the UKARA research, it is indicated that the accuracy of the prediction of correct and false answers is not only influenced by the model. The accuracy of predictions greatly affects the dataset used. In this study, the dataset is still small and not balanced for the training process. In fact, datasets are very important in machine learning, the more datasets used in the training process, the better the model produced. Adding datasets can be done in two ways, namely adding data sets manually or adding data with the data augmentation process. The process of adding datasets manually takes a long time and must go through a long process, while the process of data augmentation can be done quickly through the system.

This study introduced a data augmentation method, namely EDA (Easy Data Augmentation Techniques) which was previously studied in English literature [7]. In this study, EDA was applied in Indonesian using the UKARA dataset by doing several developments such as using BERT word embedding and using posttagging in it.

2. METHODS

The research process of automatic essay scoring using data augmentation in bahasa indonesia in general begins with the process of prepocessing, augmentation, implementation of k-fold cross validation, implementation of BiLSTM with Fasttext, and model evaluation. Here's the architecture of the study:

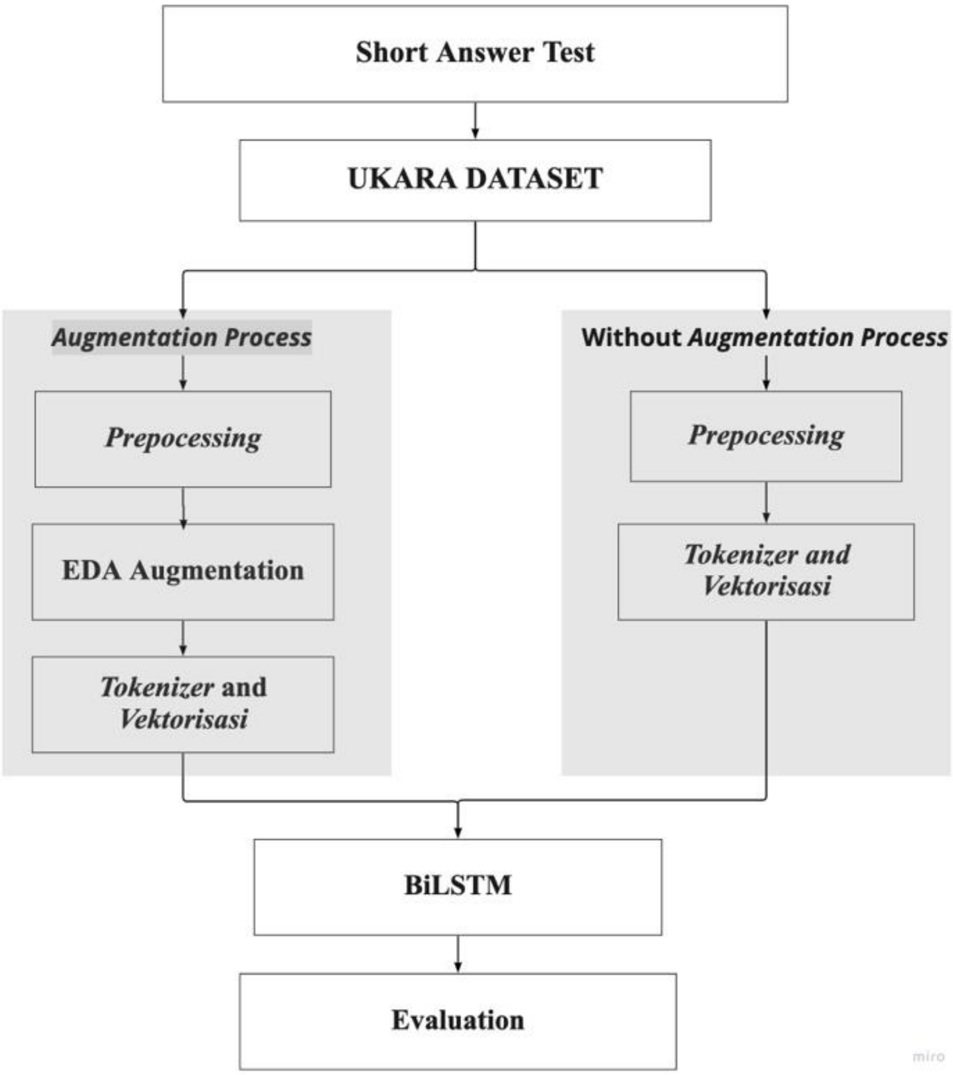

Figure 1. Research Architecture

2.1 Prepocessing

The prepocessing process is divided into several parts, namely filtering to remove special characters, such as commas (,), periods (.), exclamation marks (!), quotation marks ("), hastags (#), question marks (?), and other characters: $, %, &, ', ', *, +, -, /, :, ;, <, =, >, @, [, \,], ^, _, ', {, |, }, ~. Case folding to convert uppercase characters to lowercase. Tokenization is the process of splitting sentences into words collected in an array. And padding menyakan Length of input sentences. The word embedding application is to convert the list in the array into a vector that can be read by a computer.

2.2 EDA Augmentation Process

The EDA augmentation process in general has 4 methods, namely Synonym Replacement (SR), Random Insertion (RI), Random Swab (RS), and Random Deletion (RD). The details of this process:

2.2.1 Synonym Replacement (SR)

SR is a method of creating new data by randomly selecting words in sentences and then replacing them with synonyms of the selected word. In the study [7] the SR method was carried out by: Taking words in one sentence at random, Replacing synonyms of words that have been randomly selected, and Inserting synonyms of selected words to the position of words that have been taken randomly using the corpus provided by the NLTK library. Since the list of Indonesian vocabulary in NLTK is still limited, in this study the author made two modifications in the implementation process. Such modifications are:

1. Synonym Replacement (SR) with Embedding Indobert, The implementation of the program begins with installing the transformer and torch libraries. Then download Indobert tokenizer and Indobert Model Mask in Indonesian. After that the process to call the data to be segmented, and finally apply the SR algorithm
2. Synonym Replacemant (SR) with Embedding Indobert and Postagging The embedding implementation at this stage is the same as the SR stage, it's just that, words are randomly selected based on postagging or randomly selected words are words that have an important function in sentences

2.2.2 Random Insertion (RS)

The RS process is almost the same as SR with embedding, except that in SR with embedding the randomly selected word is replaced with the word 'MASK', while in RS the word 'MASK' is inserted randomly in the sentence. The next process is the same as the process in SR.

2.2.3 Random Deletion (RD)

Random deletion (RD) is the process of augmentation of data by randomly deleting one of the words in a sentence to produce a new sentence. In the random deletion section, the author modified the EDA RD method by adding a postagging process in it. Postagging aims to ensure randomly deleted words are not verbs, nouns, adverbs, and adjectives. So that even if there is a word that is deleted in the sentence, the sentence does not lose its meaning in meaning.

2.3 K-Fold Cross Validation

K-Fold Cross Validation is one of the methods for evaluating classifier performance, in cross validation is known as rotational estimation by dividing the data into k-sets of almost the same size, the model is classified as trained and tested as much as k each loop, part of the data is used as training data and part of it is used as test data.

2.4 Bidirectional Long Sort Term Memory (BiLSTM)

The implementation of the deep learning process with BiLSTM in this study uses tanserflow and keras libraries. A library provided by the python programming language for machine learning and deep learning BiLSTM. Before entering the BiLSTM model, there is the addition of an embedding model whose function is to insert a fasttext model and vectorize data based on the program in the embedding section described earlier. Next is to perform tuning.

2.5 Evaluation

At this stage, the system will be evaluated using a confusion matrix. There are four values generated in the confusion matrix table, between True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN). True Positive (TP). The resulting evaluation metrics are Accurasy, Precision, Recall, and F1-Score or F-measure.

3. RESULTS AND DISCUSSION

This section will discuss the augmentation results obtained using augmentation and the results of training data comparison of all models that have gone through the data training process using the BiLSTM model.

3.1 Data

UKARA provides two different data with data distribution as shown below:

Table 1 Ukara Data Distribution

Data	Total of Data	Label of Correct	Label of False
Training_A	854	609	245
Val_A	215	153	62
Test_A	268	191	77
Training_B	974	531	437
Val_B	244	135	109
Test_B	305	168	137

The training data in the two data then goes through an augmentation process so as to produce data according to table 2.

Table 2. Data Distribution After Augmentation Process

Model	Data A			Data B		
	Total	Correct	False	Total	Correct	False
Without Augmentation	854	609	254	974	531	437
EDA_Sinonim_Indobert	1624	1188	436	1921	1074	848
EDA_Sinonim_Indobert_Tag	1536	1152	384	1770	1024	746
EDA_Delete	1588	1164	422	1920	1064	846
EDA_Insert	1624	1188	436	1922	1074	848
EDA_Swab	1708	1218	490	1948	1074	874

In the application of k-fold cross validation, the training and validation data are combined first before being augmented so that a distribution of data is generated like table 3.

Table 3 Data Distribution After K-Fold Cross Validation Data Augmentation Process

Model	Data A			Data B		
	Total	Correct	False	Total	Correct	False
Without Augmentation	1069	763	307	1218	672	546
SMOTE	2032	762	762	1344	672	672
Borderline-SMOTE	2032	762	762	1344	672	672
EDA_Sinonim_Indobert	2030	1494	536	2404	1342	1062
EDA_Sinonim_Indobert_Tag	1923	1452	472	2222	1278	944
EDA_Delete	1978	1458	520	2384	1328	1056
EDA_Insert	2030	1494	536	2404	1342	1062
EDA_Swab	2138	1524	614	2438	1344	1092

Based on the distribution of data in table 2 and table 3 above, it can be seen that the results of EDA augmentation have an unbalanced distribution of data. This is caused by the augmentation process which only relies on the number of sentence rows in the data, regardless of the label of the data.

3.2 Augmentation Results

3.2.1 Synonym Replacement (SR) with Indobert Embedding

Table 4 Results of SR Augmentation with indobert embedding

Input	mereka perlu menyesuaikan diri dan beradaptasi dengan lingkungan yang baru
Masking	mereka perlu [MASK] diri dan beradaptasi dengan lingkungan yang baru
Output	mereka perlu bela diri dan beradaptasi dengan lingkungan yang baru

From the table above, it can be seen that embedding indobert to find the right word equivalent can indeed be used, but if it is used to find word synonyms, this model is still not good to use. For example sentences ‘komunitas dan negara’ which then word ‘negara’ produces synonyms ‘gas’. When studied in context ‘komunitas dan gas’ does have meaning, but when compared to input sentences ‘komunitas dan negara’ this sentence has a different meaning.

3.2.2 Synonym Replacemant (SR) with embedding Indobert dan postagging

Table 5 Results of Augmentation of Synonym Replacement (SR) with indobert Embedding and Postagging

Input	Mereka akan sulit beradaptasi dengan keadaan iklim di daerah baru yang mereka datangi.
Masking	Mereka akan [MASK] beradaptasi dengan keadaan iklim di daerah baru yang mereka datangi.
Output	mereka akan dapat beradaptasi dengan keadaan iklim di daerah baru yang mereka datangi

Based on the results above, it can be concluded that the choice of words based on tagging or choosing words that are important to replace ‘MASK’ is not the right choice, because some word replacements will actually eliminate the true meaning of the sentence. For example, on inputs ‘Mereka akan sulit beradaptasi dengan keadaan iklim di daerah baru yang mereka datangi’ with the chosen word for mask is ‘sulit’. This difficult word turned out to be replaced by the word ‘dapat’, so that the sentence turns into ‘mereka akan dapat beradaptasi dengan keadaan iklim di daerah baru yang mereka datangi’. After being traced, the meaning of the sentence is precisely the opposite of the meaning of the input sentence.

3.2.3 Random Insertion (RI)

Table 6 Results of Augmentation Random Insertion

Input	tanah yang ditempatinya, membangun rumah lagi
Masking	tanah yang [MASK] ditempatinya membangun rumah lagi
Output	tanah yang harus ditempatinya membangun rumah lagi

What is shown above are some examples of data augmentation results with the random insertion method. Based on the author’s observations of the data of this type of augmentation results the prediction of the word crucified is quite good, although many of the words inserted in the sentence are connecting words and adverbs such as atau, ini, juga, di, yang, a.

3.2.2 Random Deletion (RD)

Table 7 Results of Random Deletion Augmentation

Input	kehilangan lahan tempat tinggal dan lahan pertaniaan
Kata yang dihapus	dan
Output	kehilangan lahan tempat tinggal lahan pertaniaan

Based on the table above and the author’s observations of the resulting dataset, the results of data augmentation using RD are quite good, although sometimes words that are considered unimportant turn out to have a big impact in a sentence, for example in sentences ‘kehilangan lahan tempat tinggal lahan pertaniaan’ which is then omitted the word ‘dan’ in it will be ‘kehilangan lahan tempat tinggal lahan pertaniaan’. Word ‘dan’ in the input sentence it is emphasized that the object in question is two in number, namely the land of residence and agriculture. While the meaning after the word ‘dan’ in the input sentence it is emphasized that the object in question is two in number, namely the land of residence and agriculture. While the meaning after the word

3.2.2 Random Swab (RS)

Table 8 Random Swab Augmentation Results

Input	harus beradaptasi dengan lingkungan sekitar, harus bertahan hidup ditempat tinggal yang baru.
Kata yang ditukar	Dengan dan ditempat
Output	harus beradaptasi dengan ditempat sekitar harus bertahan hidup lingkungan tinggal yang baru

Based on the results shown in table 6.13 it can be concluded that augmentation data using RS can change the meaning of the previous word, as in the input sentence ‘harus beradaptasi dengan lingkungan sekitar, harus bertahan hidup ditempat tinggal yang baru’ which then turns into ‘harus beradaptasi dengan ditempat sekitar harus bertahan hidup lingkungan tinggal yang baru’ the augmentational sentence becomes very difficult to understand even by humans.

3.3 Hypertuning Parameter

This method aims to see the best models of the architecture. The test was performed using the BiLSTM architecture with Fasttext embedding. The goal is to find out the F1 score results of each of the best models after tuning. The results of the tuning parameters obtained are that the number of epochs suitable for use for data A is 35 epochs with an accuracy value of 83.58% and F1 score is 83.70, while for data B is 40 epochs with an accuracy of 68.19 and an F1 score value is 67.44. Thus achieved some tuning that is suitable for both data, namely data A number of layers = 150, L2 regulation = 0.0001 and epoch = 35, while for data B, the number of layers = 50, L2 regulation = 0.001 and epoch = 40.

3.4 Model Evaluation Results

The test results of each augmentation data for data A can be seen in the figure

Table 9 Results of Data Model A Evaluation

Augmentation		**Time (minutes)**	**Accuracy (%)**	**F1 Score (%)**	**Precissi on (%)**	**Recall (%)**
Without Augmentation	Not K-fold	**16.77**	**83.95**	**84.10**	**84.31**	**83.95**
	With K-fold	**21.76**	**85.07**	**84.81**	**84.74**	**85.07**
EDA_Sinonim_ Indobert	Not K-fold	**28.36**	**84.70**	**84.73**	**84.76**	**84.70**
	With K-fold	20.28	83.20	83.03	82.92	83.20
EDA_Sinonim_ Indobert_Tag	Not K-fold	30.08	82.08	82.01	81.95	82.08
	With K-fold	29.76	83.58	82.76	83.24	83.58
EDA_Delete	Not K-fold	32.07	81.71	82.00	82.47	81.71
	With K-fold	17.69	83.58	82.86	83.09	83.58
EDA_Insert	Not K-fold	33.87	81.71	80.74	81.05	81.71
	With K-fold	30.56	83.20	82.79	82.72	83.20
EDA_Swap	Not K-fold	33.91	81.71	81.68	81.64	81.71
	With K-fold	34.90	83.58	83.22	83.13	83.58

Based on the figure above, it can be concluded that from several results of the training process and data evaluation without augmentation with k-fold cross validation has the highest accuracy of 85.07% followed by EDA synonym indobert without k-fold cross validation with an accuracy value of 84.70%, and finally followed by data without augmentation and not using k-fold cross validation with an accuracy of 83.95%. The results of dataset B are listed in table 10

Table 10 Results of Data Model Evaluation B

Augmentation		**Time (minutes)**	**Accuracy (%)**	**F1 Score (%)**	**Precission (%)**	**Recall (%)**
Tanpa Augmentasi	Not K-fold	10.92	68.85	68.69	68.71	68.85
	With K-fold	**09.07**	**72.45**	**71.89**	**72.80**	**72.45**
EDA_Sinonim_Indobert	Not K-fold	**16.23**	**72.13**	**71.19**	**73.10**	**72.13**
	With K-fold	**17.40**	**72.13**	**71.95**	**72.04**	**72.13**
EDA_Sinonim_Indobert_Tag	Not K-fold	15.89	69.83	69.44	69.80	69.83
	With K-fold	14.08	71.47	71.36	71.37	71.47
EDA_Delete	Not K-fold	16.43	68.52	66.84	70.00	68.52
	With K-fold	16.18	71.80	71.22	72.11	71.80
EDA_Insert	Not K-fold	16.38	62.29	62.35	63.25	62.29
	With K-fold	**14.01**	**72.78**	**72.50**	**72.79**	**72.78**
EDA_Swab	Not K-fold	18.50	68.19	68.18	68.17	68.19
	With K-fold	16.98	70.16	69.58	70.33	70.16

For Data B, the highest accuracy value was obtained by EDA insert data with k-fold cross validation of 72.78%, followed by data without augmentation using k-fold cross validation of 72.45% and the third rank was obtained by EDA synonym indobert using *k-fold cross validation* and not with the same accuracy value of 72.13%.

4. CONCLUSIONS

The main purpose of this article is to analyze the effect of automatically adding datasets using the Easy Data Augmentation (EDA) augmentation method on increasing the accuracy value and f-score of BiLSTM deep learning with Fasttext on the short description test scoring system using the UKARA dataset. The results showed that in dataset A using k-fold cross validation, the highest accuracy value with a value of 85.07% was followed by the synonym EDA indobert without k-fold cross validation with an accuracy value of 84.70%, and finally followed by the synonym EDA indobert without k-fold cross validation with an accuracy value of 84.70%, and finally followed by data without augmentation and not using k-fold cross validation with an accuracy of 83.95%.

The results in data B showed EDA insert with k-fold cross validation of 72.78%, followed by data without augmentation using k-fold cross validation of 72.45% and the third rank was obtained by the synonym EDA indobert using k-fold cross validation and not with the same accuracy value of 72.13%. So it can be concluded that in data A there is no increase in accuracy when using the EDA augmentation method while in data B it obtains an increase in accuracy when using the EDA augmentation method. It is hoped that further research will be explored by replacing the indobert embedding method with other methods, balancing the results of augmentation data before conducting model training, adding a combination of tuning parameters to the BiLSTM Fasttext model, and making comparisons with other methods such as back translation and semantic similarities.

REFERENCES

[1] Mansyur, S., & Harun, R. (2015). Asesmen pembelajaran di sekolah: Panduan bagi guru dan calon guru. Yogyakarta: Pustaka Pelajar.

[2] Arikunto Suharsimi. (2013). Dasar-Dasar Pendidikan (2nd ed.). Bumi Aksara.

[3] Ruslan, R., Gunawan, G., & Tjandra, S. (2018, August). Sistem Penilaian Otomatis Jawaban Esai Menggunakan Metode GLSA. In Seminar Nasional Aplikasi Teknologi Informasi (SNATi).

[4] Fadilah, N. (2016). Rancang Bangun Sistem Penilaian Tes Essai Berbasis WEB di Testing Center UNM. Universitas Negeri Makassar

[5] Herwanto, G. B., Sari, Y., Prastowo, B. N., Bustoni, I. A., & Hidayatulloh, I. (2018). UKARA: A fast and simple automatic short answer scoring system for Bahasa Indonesia. ICEAP 2019, 2, 48-53.

[6] Purwarianti, A. (2019, October). Effective Use of Augmentation Degree and Language Model for Synonym-based Text Augmentation on Indonesian Text Classification. In 2019 International Conference on Advanced Computer Science and information Systems (ICACSIS) (pp. 217-222). IEEE.

[7] Wei, J., & Zou, K. (2019). Eda: Easy data augmentation techniques for boosting performance on text classification tasks. arXiv preprint arXiv:1901.11196.