IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Revue d'Intelligence Artificielle
Vol. 37, No. 5, October, 2023, pp. 1101-1108
Journal homepage: http://iieta.org/journals/ria

Evaluating the Impact of Sentence Tokenization on Indonesian Automated Essay Scoring Using Pretrained Sentence Embeddings

Nurul Chamidah*, Evi Yulianti, Indra Budi

Faculty of Computer Science, Universitas Indonesia, Depok 16424, Indonesia

Corresponding Author Email: nurul.chamidah@upnvj.ac.id

https://doi.org/10.18280/ria.370502

Received: 9 June 2023
Revised: 21 July 2023
Accepted: 29 July 2023
Available online: 31 October 2023

Keywords:
automated essay scoring, Indonesian, sentence embeddings, sentence tokenization, Siamese Manhattan LSTM

ABSTRACT

Automated Essay Scoring (AES) systems are designed to expedite the assessment process, where human scoring is frequently slow and subject to inconsistencies and inaccuracies. This study, therefore, investigates the role of sentence tokenization in the performance of Indonesian Automated Essay Scoring, given that Natural Language Processing (NLP) techniques are requisite in AES to handle student responses that present identical semantic meanings but vary in length. A distinct approach was adopted in which full answers were not vectorized; instead, they were fragmented into sentences prior to vectorization. This method was deemed potentially more effective due to the high probability of discrepancies in sentence order between reference and student responses. Sentence embeddings, which encapsulate a sentence as a sole vector, were utilized. Pretrained SBERT-based sentence embeddings were employed to vectorize sentences from both reference answers and student responses, serving as semantic features for the Siamese Manhattan LSTM (MaLSTM) model. The MaLSTM model possesses the ability to process two inputs and evaluate their similarity using the Manhattan distance metric and use this similarity value as a predictive scoring output. This score was subsequently compared to human scores using the Root Mean Square Error (RMSE) and Pearson Correlation. Interestingly, sentence embeddings without tokenization slightly outperformed those with sentence splitting, as evidenced by a 0.61% improvement in RMSE and a 0.01 increase in Pearson Correlation. The results obtained indicate that sentence tokenization, as applied to the Indonesian Automated Essay Scoring dataset, does not have a notable impact on essay scoring performance. Therefore, it may be concluded that the application of sentence tokenization is not a necessary step in this dataset's text-processing phase of AES.

1. INTRODUCTION

Automated Essay Scoring (AES) represents a critical task in Natural Language Processing (NLP), offering substantial utility in evaluating learning or examination methodologies that involve essay-type questions. In such question types, assessment focuses on the content of the answers, wherein students provide responses ranging from a single phrase to a paragraph by drawing upon external knowledge [1].

Objective-based questions, wherein students select from predefined options, present a simpler task for assessment compared to essay-type responses in natural language. Given the predetermined nature of the answers for objective questions, constructing an automated scoring system for such questions is relatively straightforward. Conversely, for responses expressed in natural language, evaluations must be conducted with a focus on the content of the written text, necessitating any automated scoring system to possess a comprehension of natural language.

Questions that require essay-type answers based on a reference necessitate comparing the student's response with a reference answer in natural language. This comparison process can be intricate, as it is not merely the words utilized in the answers that are compared but rather the content contained within them. As the responses are expressed in natural language, how the answers are articulated may vary, both in terms of the lexicon employed and the length of the responses. It is plausible that the terminology used within a reference answer could diverge from that in a student's answer, yet the latter may still be correct. Additionally, the text length of the reference and student's answers may differ, highlighting the necessity for an assessment technique that can effectively match the content of the reference and student's answers.

One strategy for comparing the content of answers involves assessing the semantic similarities between the reference and student's answers. A higher degree of semantic similarity between a student's response and the reference answer would correspond to a higher score. This literature review and introduction outline the complex nature of AES and the need for effective strategies in comparing answer content, setting the stage for the investigations presented in this study.

1.1 BERT-based embeddings

BERT is one of the highly utilized language models that is constructed using a bidirectional model, extensive corpora, and encoder transformers to develop language models for tasks involving semantic textual similarity, as described in the study [2]. BERT vectors are utilized to represent the word level. However, when comparing two sentences to determine semantic similarity, sentence vectors created from BERT vectors are ineffective. Hence, Reimers and Gurevych [3]

developed SBERT, a sentence-level language model of BERT. This model employs siamese BERT and triplet networks to generate sentence embeddings that are semantically meaningful.

Studies on automated essay scoring based on semantics were carried out by researchers before. Research [4] on the Introduction to Networking Computer Science subject used semantic feature SBERT sentence embeddings to vectorize 228 student answers and their reference answers, then the two vectors were compared with Cosine Similarity. A similar study [5] compared some of the semantic features of pretrained SBERT for assessing short answers in the Mohler dataset [6], where Cosine Similarity determined the score. Their study shows that SBERT performs better than using GloVe embeddings and Siamese BiLSTM combined. SBERT shows good performance based on these studies. So, in this study, we use SBERT as sentence embeddings for automated essay Scoring to transform text answers into vectors.

The out-of-length text also became a consideration in some studies. A study [7] utilizes BERT word embeddings as semantic features in automated essay scoring by paying attention to the length of the input sequence in the BERT model. In this study, they used a hierarchical model so that out-of-length input sequences are not truncated. Instead, input sequences become the input to the hierarchical BERT model and produce better performance than the truncated input sequence. While word embeddings are used to vectorize word-by-word text, full-text embeddings such as sentences or paragraph embeddings are obtained by summarizing its vector. For example, Lubis et al. [8] use a dataset [9] utilizing semantic features obtained from a combination of POS Tags and word embeddings Word2Vec. They average word embeddings vectors as text embeddings in reference and students' answers. Hence, we employ averaging in this study to get paragraph embeddings from sentence embeddings SBERT.

1.2 LSTM-based automated scoring

The employment of a deep learning-based method in automated scoring has shown the most promising results, as proven by the study [10]. Study [11] employs a Long Short Term Memory (LSTM)-based model, utilizing Manhattan distance as similarity in the output layer. Two tokenized and vectorized sentences are fed to the Siamese LSTM network, with the output calculated in the output layer using Manhattan distance as the similarity score. Similarly, in their research, Mueller and Thyagarajan [12] also utilize Siamese LSTM models to compute sentence similarity, using the SICK dataset of the SemEval 2014, with word embeddings Word2Vec [13]. Furthermore, their research was continued in [14] to assess short answers and predict scores using the dataset in Indonesian.

Meanwhile, the study [15] uses semantic features to vectorize, employing SemSpace embedding. Subsequently, it is trained with Siamese Manhattan LSTM on the Mohler and CU-NLP datasets, showcasing state-of-the-art results. Based on these studies, Siamese Manhattan LSTM has proven to provide good performance. Therefore, in this study, we employ Siamese Manhattan LSTM as the model to predict scores in automated essay scoring [15].

Based on these studies, we proposed an automated essay scoring in the dataset [16, 17] using the semantic features of sentence embeddings SBERT with sequence length handling by averaging as in the previous study [8]. The sentence embeddings transform text answers into vectors as features that can be processed further. Students' and reference answers embedding vectors were then trained using Siamese Manhattan LSTM model [14, 15], which has proven that Siamese Manhattan LSTM has good performance. The Siamese Manhattan LSTM acts as a score predictor that learns from some portion of the dataset, and then the trained model can predict scores from new incoming data.

We propose sentence tokenization because the arrangement of sentences in a paragraph with the same semantic meaning can be different, in this case, the student's answers and reference answers. Therefore, breaking down the sentence first and then vectorizing and averaging its vectors is expected to represent better the student answer's vector and the reference answer's vector. This study aims to evaluate and understand the effect of sentence tokenization on automated essay scoring performance using sentence embeddings SBERT as semantic features and Siamese Manhattan LSTM model to generate scores.

2. RELATED WORK

Online learning has been widely applied in learning since the Covid-19 pandemic as a medium for providing learning materials to evaluate learning, such as exams. The exam, a form of evaluation in the learning process, has various types of questions that can be classified into written (such as essay and short answer) and objective questions. Objective questions provide answer choices such as matching, true-false choice, or multiple choice. Meanwhile, written answer questions do not provide answer choices, where the student have to write their own answer sentences, for example, describing, giving arguments, and explaining.

Essay questions are widely used because they are considered better at measuring student understanding of a lesson than objective questions [18]. Assessment of objective questions is more effortless than evaluating answers from essay questions because it can be optimized with a system with definite answers. Unlike essay answers, which are more expensive to assess because human evaluators have to be evaluated manually, it takes longer to evaluate essays, and there are consistency problems. This consistency problem arises because, in essay questions, answers are written according to the understanding or knowledge of the students with their writing style and delivery, which causes the evaluator's interpretation to vary.

The problem of consistency in giving scores to students' answers can arise from the same evaluator or different evaluators [19]. Inconsistent assessment of answers from the same evaluator often occurs when evaluations are carried out at different times, whereas other evaluators can also provide their subjectively different scores.

A solution to the essay scoring problem is to build automated essay scoring. This automatic essay scoring model is very challenging because the computational model is expected to provide scores as similar as possible to human evaluator scores, where students can answer one question with various explanations. We hope that essay assessment can be carried out more objectively, following the standards or referencing answers expected to be in the answer using automated essay scoring.

2.1 Feature extraction in automated essay scoring

Features extracted or used in automated scoring studies such as lexical, syntax, and semantic features, as well as combinations of these features are already done by the researcher. Lexical features used in studies such as Bag of Words (BoW) [9] and TF-IDF [20] are handcrafted to depict essay text in AES. In other studies, lexical features representation is pretrained using shallow neural network models such as Word2Vec [21], GloVe [22], and Fasttext [23]. Pretrained embedding models constructed through deep learning methodologies offer an effective solution for numerous Natural Language Processing (NLP) tasks that are trained using large corpora. A study conducted by Gaddipati et al. [24] employed BERT, GPT-2, and ELMo to facilitate text similarity analysis between the answers provided by students and the reference answers. Another work [25] compared BERT and XLNET by excluding questions from the training input. Study [7] used word embeddings BERT for scoring and combining semantic features BERT with syntax features such as POS Tags, sentence length, and the ratio of unique words to the number of words [26]. Sentence BERT (SBERT) endeavored to address BERT's computational overhead in the context of textual semantic similarity analysis between two sentences by incorporating Siamese BERT and triplet networks introduced by Reimers and Gurevych [3]. This approach is employed in investigations [4, 5], where SBERT vectors are utilized as a semantic attribute to conduct a comparative examination between students' responses and the reference answers, aiming to determine the students' scores using Cosine Similarity.

2.2 Machine learning techniques in automated essay scoring

Automated scoring techniques for assessing essays as reference-based questions, mainly performed by comparing reference and student's answer texts. In the existing studies, scoring is accomplished by employing machine learning and similarity or distance techniques. Some investigation utilizes similarity techniques, i.e., using Cosine Similarity to compute similarity by comparing reference and students' answers from their word embeddings [8] and sentence embeddings [5]. The scoring technique based on lexical similarity compares texts using Cosine Similarity, Euclidean Distance, and Jaccard Coefficient [17]. Study [9] uses Bag of Words features where student answers and answer keys are matched based on the number of words that match, Cosine Coefficient, Jaccard Coefficient, and Dice Coefficient. Lubis et al. [8] continued this study, which used semantic similarity features by combining word2vec with POS Tag, and then scoring was carried out by Cosine Similarity. Another study used Cosine Similarity and semantic features [5], but this study focuses on comparing SBERT-based sentence embeddings as semantic features. Whereas machine learning techniques i.e., Multi-Layer Perceptron (MLP) [27, 28], Linear Regression [26], SVM and KNN [15, 29, 30].

Besides traditional machine learning, the growth of deep learning methodologies have given significant development to the exploration of sentence-level semantic similarity through utilizing LSTM-based networks. This approach employs LSTM to measure the semantic similarity between two texts, one of which employs the Siamese Manhattan LSTM approach introduced by Mueller and Thyagarajan [12]. This technique measures the similarity of two texts by employing Manhattan distance on the output of two identical LSTMs. Study [14, 15] utilized this Siamese LSTM model to compare student answer texts with reference answers in automatic assessment, yielding commendable performance.

The investigation [14] aims to conduct an assessment of short answers and predict scores employing the Indonesian short answer dataset through Word2Vec as word embeddings and Siamese Manhattan LSTM. Another study carried out by Tulu et al. [15] also utilized Siamese Manhattan LSTM and Semspace sense vectors. These studies employed word-level embeddings to vectorize students' answers and reference answers. However, in this study, we propose employing sentence-level embeddings using pretrained SBERT, which has demonstrated good performance in previous works [4, 5]. It is important to highlight that our study is different from the approach taken by authors [4, 5] that rather than utilize deep learning to predict scores, they fine-tuned SBERT and calculated its scores using Cosine similarity. Furthermore, we aim to evaluate the impact of sentence tokenization in automated essay scoring, an aspect that has not been previously explored.

Our investigation's objectives are focused on comparing the impact of sentence tokenization on the efficacy of automated essay scoring. In this study, we propose automated essay scoring that utilizes pretrained sentence embeddings SBERT and Siamese Manhattan LSTM to assess student responses based on reference answers and generate scores accordingly. The dataset employed in this study is the Indonesian automated essay scoring by Roshinta et al. [16, 17]. The textual answer will undergo sentence tokenization and subsequently be vectorized utilizing pretrained SBERT. The scores output from Siamese Manhattan LSTM will be evaluated against human scores to gauge the performance of the abovementioned method.

3. METHODOLOGY

The research methodology of this study is illustrated in Figure 1. Reference answers and students' answer text in the dataset are preprocessed and split to obtain sentence tokenization. Sentence embeddings then vectorize every sentence in the paragraph. Final embedding of paragraph obtained by averaging its vectors. These vectors are inputs for Siamese Manhattan LSTM. The model outputs are then converted to scores and evaluated to measure the model's performance.

3.1 Dataset

The dataset used in this study is from the research [16, 17], which can be downloaded freely at Mendeley data. There are explanation questions with four categories (politics, sports, lifestyles, and technology) in Indonesian. Each category has ten questions and reference answers corresponding to the question with a total of 40 reference answers for all categories. A total of 2162 students' answers are available with minimum length of 1, maximum of 4259, the average length of 187 characters, and three scores that evaluators give manually in the range of 0 to 100 for each answer. Therefore, in this study, we used average scores from these three evaluators as gold standards and normalized to the range [0, 1].

3.2 Preprocessing

Reference answers and student's answers text preprocessed by case folding, punctuation removal, and drop record if the student's answer length is less than two characters. Case folding is used to transform all characters to their lowercase form. Punctuation removal to eliminate punctuation marks such as slashes, quotation marks, and question marks. Additionally, answers containing fewer than two characters, such as single characters or single letters, empty answers because of punctuation removal are disregarded and dropped as they are considered gibberish responses in the student's answer. Sentence tokenization splits sentences by character stop/dot (.) and hyphen (-). After preprocessing, the data used in this study are reduced from 2162 to 2157 records.

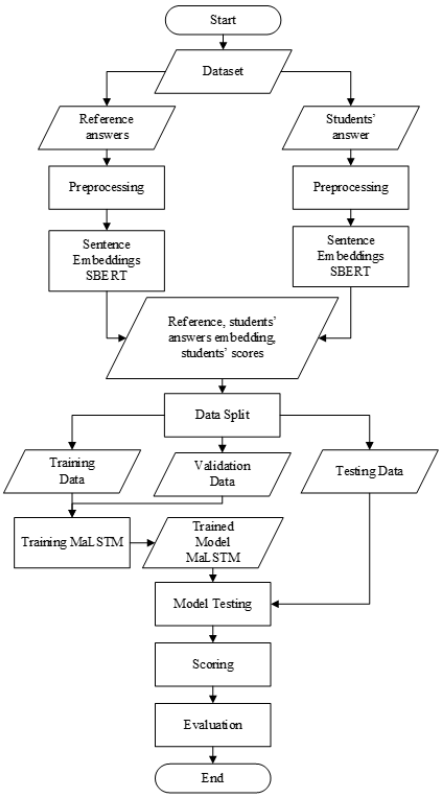

Figure 1. Research methodology

3.3 Sentence embeddings SBERT

SBERT is pretrained sentence embeddings developed using Siamese BERT [3]. SBERT is modified from BERT, which uses triplet and Siamese networks to generate sentence embeddings that have semantic meanings. The Siamese network is constructed using two similar networks that share weights. The SBERT architecture is illustrated in Figure 2, wherein the network receives two input sentences and subsequently generates sentence embeddings. The two sentence vectors produced by this network are then compared to acquire a similarity score using cosine similarity.

There are existing pre-trained sentence transformer models available that are trained on different corpora for different tasks. An overview of the pre-trained SBERTs is provided in Table 1. In order to obtain the optimal model, we conducted experiments concerning various dimensions and maximum sequence lengths of pretrained SBERT SBERTs to perform sentence embedding in this study.

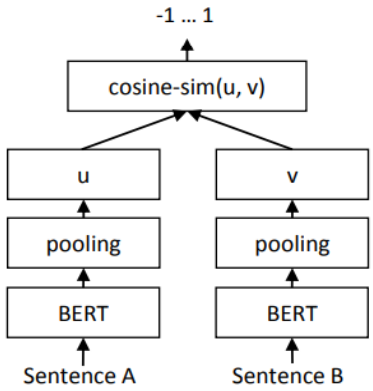

Figure 2. SBERT architecture [22]

Table 1. Pretrained SBERT model

Pretrained Model	Max Sequence Length	Dimension
distiluse-base-multilingual-cased-v2	128	512
all-distilroberta-v1	512	768
multi-qa-distilbert-cos-v1	512	768
nli-distilroberta-base-v2	75	768
all-MiniLM-L6-v2	256	384
paraphrase-albert-small-v2	256	768

Pretrained SBERT vectorizes sentences or paragraphs in reference answers and students' answers. There are two scenario embeddings: firstly, for the entire text in answers, and secondly, for each split sentence in a paragraph. From split sentence embeddings, final embeddings for text answers are calculated by averaging from these split sentence embedding vectors. Each embedding result is then combined as a resource containing reference answer embeddings, students' answer embeddings, and scores.

3.4 Data split

We split 2157 records data as training data 70% in order to enable a model to make accurate predictions on new data. It is imperative that a significant portion of the training data is utilized. This is due to the fact that the model must learn various patterns present in the dataset for the model to be able to make generalizations. The rest of the dataset then used as validation and testing data equally, with validation data 15%, and testing data 15%.

Training and validation data are used to train and build a deep learning model. Testing data are used to evaluate model and ensure that testing data is never used in the training process. In this splitting data phase, we use stratified sampling to ensure that every question is presented in training, validation, and testing data. So that there are no unseen questions, we focus only on unseen student answers.

3.5 Siamese Manhattan LSTM

Siamese Manhattan LSTM, as proposed by Mueller and Thyagarajan [12], is utilized for assessing the semantic similarity between two texts by applying the Manhattan distance to the output of two identical LSTMs. In the context of automated essay scoring, Siamese networks are fed with two input vectors: students' answers and reference answers, which are subsequently trained with human scores as the

actual output. The trained Siamese LSTM model is then implemented for predicting scores when both the student's answer and the reference answer are fed into the model.

Siamese Manhattan LSTM is a model that implements identical LSTM, LSTMa, and LSTMb in parallel [12]. In this study, Siamese MaLSTM can be seen in Figure 3. We used identical LSTMs, $LSTM_k$ for reference answers, and $LSTM_s$ for students' answers.

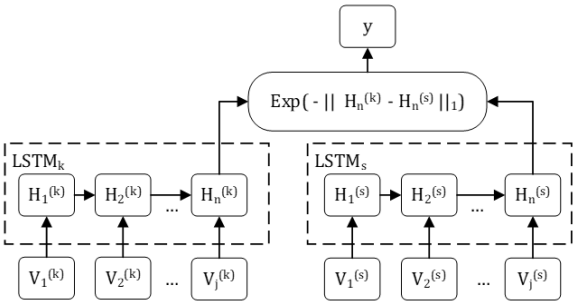

Figure 3. Training model

LSTM's input is the vector (V) with the length of j, which is the length of pretrained SBERT embeddings obtained from splitting paragraphs into sentences and averaging its sentence embeddings. These two LSTMs have the same hidden neurons (H) number of n=150. Finally, the output of $LSTM_k$ and $LSTM_s$ is calculated using Manhattan distance as y as seen in Eq. (1). Gold standard output or actual output (y_{act}) for this model is the average of scores from three teachers that can be seen in Eq. (2). We used batch size 63, Adam optimizer, and epoch of 25.

$$y = Exp\left(-\left\|H_n^{(k)} - H_n^{(s)}\right\|_1\right) \quad (1)$$

$$y_{act} = \frac{average(scores)}{100} \quad (2)$$

3.6 Scoring and evaluation

The output of MaLSTM y in the range of 0 to 1 is multiplied by the maximum score to obtain the predicted score y_{pred} calculated by Eq. (3).

$$y_{pred} = y * \max\_score \quad (3)$$

The metric to compare the actual score and model predicted score is Root Mean Squared Error (RMSE). The Root Mean Squared Error (RMSE) metric evaluates the absolute deviation that exists between the predicted and expected scores. Consequently, a model that generates a lesser RMSE value is considered to have better evaluation performance. The equation of RMSE can be seen in Eq. (4).

$$RMSE = \sqrt{\frac{\sum(y_{actual} - y_{pred})^2}{n}} \quad (4)$$

Pearson Correlation is also used to evaluate the model's success. The Pearson correlation is a statistical metric that represents the degree of concurrence between two variables. Specifically, in this study, the Pearson correlation is employed to measure agreement between human-assigned scores and MaLSTM generated scores. The higher correlation between predicted score and actual score, the better model performance. Pearson Correlation coefficient computed by Eq. (5). Success criteria for essay scoring based on correlation value is excellent if $corr > 0.75$, Good if $corr$ between $0.40 - 0.75$, and poor if $corr < 0.4$.

$$corr(y_{pred}, y_{act}) = \frac{cov(y_{pred}, y_{act})}{stdev(y_{pred}).stdev(y_{act})} \quad (5)$$

Limitations and potential biases in the methodology used in this study that sentence tokenization is only applied to both reference and students' answers, we did not conduct separation between answers that should consist of some sentences or clauses based on questions such as listing questions, and which questions that should consist of single sentences. Moreover, we only trained MaLSTM for scoring, pretrained SBERT model was used without fine tuning or transfer learning. In this case, we only compare sentence tokenization implications using the existing SBERT model.

4. RESULT AND DISCUSSION

MaLSTM model for automated essay scoring has been experimented on the dataset [16, 17]. There were 2157 students' answers from 40 questions processed. The model was trained using 70% of the data and validated using 15%, and the remaining 15% was used for testing. We use stratified sampling based on questions to ensure every question is present in the split data.

Model evaluation is done by comparing scores generated by MaLSTM model and the average scores given by human evaluators in testing data. These scores are compared by RMSE and Pearson Correlation metrics to measure model performance. We also compare MaLSTM to Cosine Similarity as a baseline model.

4.1 RMSE results

Table 2 shows the RMSE of testing results using MaLSTM and cosine similarity with embeddings technique by averaging embedded vector and without averaging (all text embeddings). The best results for all scenarios were achieved using pretrained distiluse-base-multilingual-cased-v2 for all models and embedding techniques.

Table 2. RMSE of testing result

Pretrained SBERT	MaLSTM		Cosine	
	Average	All	Average	All
distiluse-base-multilingual-cased-v2	11.26	10.65	23.36	18.98
all-distilroberta-v1	12.53	12.03	38.45	32.68
multi-qa-distilbert-cos-v1	12.82	12.01	39.70	36.91
nli-distilroberta-base-v2	12.93	13.19	45.05	45.04
all-MiniLM-L6-v2	12.53	12.34	36.58	31.51
paraphrase-albert-small-v2	14.61	13.36	37.02	32.92

The best automated scoring model produced by Siamese MaLSTM using all text embeddings technique without splitting sentences with RMSE 10.65, and all text embeddings achieved better RMSE in all pretrained SBERT-based except for nli-distilroberta-base-v2 where averaging technique gives better RMSE evaluation. All text embeddings without splitting sentences also give better RMSE for all pretrained SBERT in

the baseline model using Cosine Similarity. Moreover, the best performance of baseline was also achieved by pretrained distiluse-base-multilingual-cased-v2.

4.2 Pearson correlation results

Pearson Correlation between model scores and human scores also gives the same result. MaLSTM using pretrained distiluse-base-multilingual-cased-v2 by all text embeddings technique without splitting sentences gives the best result by 0.92, which is an excellent correlation, as seen in Table 3.

Table 3. Pearson correlation

Pretrained SBERT	MaLSTM		Cosine	
	Average	All	Average	All
distiluse-base-multilingual-cased-v2	0.91	0.92	0.69	0.80
all-distilroberta-v1	0.89	0.90	0.59	0.68
multi-qa-distilbert-cos-v1	0.88	0.90	0.61	0.65
nli-distilroberta-base-v2	0.88	0.87	0.55	0.59
all-MiniLM-L6-v2	0.89	0.90	0.66	0.71
paraphrase-albert-small-v2	0.84	0.87	0.58	0.64

The Pearson correlation analysis conducted on the baseline model revealed that the act of averaging sentences failed to enhance the correlation between the predicted scores and the student's actual scores. However, employing the entire answer text yielded superior results. Notably, the distiluse-base-multilingual-cased-v2, which is the best pretrained SBERT model, demonstrated a correlation of 0.80.

4.3 Example analysis

We take a closer look at the testing data with high difference scores between generated and human score. The average human score of 91 indicates a student's answer is almost correct, but the predicted score using MaLSTM is 60.09 and 61.45 (all text and average embeddings). The student gives long explanations with longer word sequences than reference answers where student's answer has 203 words and reference answer has 58 words. We suspect that long explanation sentences that outside reference answer can affect vector representing the answer and make the system gives a lower score than human's score.

Another case in question "*Sebutkan beberapa kondisi untuk dilakukannya kick off dalam sepak bola. (Sebutkan minimal 3)*" with reference answer *"-Memulai pertandingan - Terjadinya gol - Memulai babak kedua - Memulai babak perpanjangan waktu*", student's answer "- *Memulai babak pertama - Memulai babak kedua - apabila terjadi water break - apabila terjadi goal*" has average human score of 73.33, and MaLSTM 56.36 and 42.96 by embeddings averaging and all text embeddings respectively. It shows that averaging method has closer score than all text embeddings.

Another example of a student's answer "*Memulai pertandingan. Terjadinya gol. Memulai babak kedua. Memulai babak perpanjangan waktu*" has average human score of 100, this answer is considered perfect by humans, but model scores are 87.85 and 91.47 for averaging and all text embedding techniques. Averaging technique has less close score than all text embeddings to the actual score because student's answer text does not have split sentences and reference answer split by hyphens. So, all text embeddings have a more accurate score than averaging.

These sample cases show that one of the performance factors is caused by embedding techniques. Whether it is by averaging or all text embeddings, it will give better performance depending on students' writing, whether they use sentence by sentence to explain, or merge their answer in one sentence.

The potential limitation of this study lies in the diversity of the dataset. The dataset covers four distinct topics and is not domain-specific, resulting in limited answers for each question and no correlation between answers in different topics. Additionally, the dataset presents various forms of questions, such as sentences, lists of words, and lists of sentences or clauses. Consequently, there is a difference in the delivery of student answers and reference answers, which may render sentence tokenization ineffective.

The results obtained evince that the practice of sentence tokenization, as applied in the Indonesian Automated Essay Scoring dataset, does not wield any substantial impact on the efficacy of essay scoring. This is due to its inferior performance in both RMSE, which is 0.61 higher, and Pearson correlation, which is 0.01 lower, compared to without sentence tokenization. Consequently, it may be deduced that the utilization of sentence tokenization is not a prerequisite step during the text processing phase in automated essay scoring for this specific dataset.

5. CONCLUSIONS AND FUTURE WORKS

This study proposed to evaluate the impact of sentence tokenization in automated essay scoring using Indonesian essay scoring dataset [9, 10] by utilizing Siamese Manhattan LSTM and pretrained sentence embeddings SBERT. The reference and students' answers in the dataset have undergone preprocessing and sentence tokenization. Subsequently, sentence embeddings have been employed to vectorize each sentence in the paragraph. The final embedding of the paragraph has been derived by averaging its vectors and utilized as inputs for the Siamese Manhattan LSTM. The output scores of the model were then assessed to measure the model's performance. This method is compared to sentence embeddings without sentence tokenization to evaluate its performance, and whether sentence tokenization impacts automated scoring using the dataset.

5.1 Conclusions

The result shows that the best performance of sentence tokenization and averaging sentence embeddings that were trained using Siamese Manhattan LSTM is 11.26 RMSE. This RMSE result has lower performance compared to sentence embeddings without tokenization by RMSE of 10.65, which is 0.61 lower than without averaging. These RMSE results show that using sentence tokenization and then averaging its vector embeddings can lead to higher errors or larger differences between predicted scores and human scores than without sentence tokenization. Pearson Correlation that represents agreement between human score and predicted score in sentence tokenization and averaging its vectors embeddings also leads to lower agreement. Without tokenization, SBERT and MaLSTM can achieve 0.92 correlation, but with averaging, it became 0.01 lower correlation, that is 0.91. The outcomes acquired demonstrate that the act of sentence tokenization, as experimented with in the context of the Indonesian Automated

Essay Scoring dataset, does not possess any significant influence on the performance of essay scoring. As a result, one may infer that the implementation of sentence tokenization is not an essential measure during the phase of text processing in relation to automated essay scoring for this particular dataset.

5.2 Limitations and future works

The study exhibits limitations, and the suggested methodology has not achieved optimal performance. This could potentially be attributed to the dataset's heterogeneous nature, which consisting questions in various forms and diverse domains. Future research in the field of AES could be directed towards various aspects, such as expanding the dataset by incorporating additional samples. In cases where the dataset exhibits a diverse range of types, the number of samples for each type could be increased by adding samples or through alternative approaches such as data augmentation. Fine tuning or transfer learning on pretrained embeddings SBERT before performing sentence embedding could be a viable alternative. Additionally, the incorporation of other embedding techniques could be explored. Furthermore, the deep learning model employed for score prediction could be enhanced, for instance, by integrating LSTM layers to the Siamese Manhattan LSTM or utilizing other deep learning models.

The findings of this research make a significant contribution to the advancement of Automated Essay Scoring research, serving as a preliminary step towards addressing the broader issue of blended learning assessment. Further investigation is required in order to properly identify appropriate models, determine the factors that hinder model performance, and assess the effectiveness of these models on alternative Automated Essay Scoring datasets before their implementation as an assistant in the assessment process.

ACKNOWLEDGMENT

This research is funded by Directorate of Research and Development, Universitas Indonesia under Hibah PUTI 2023 (Grant No. NKB-021/UN2.RST/HKP.05.00/2023).

REFERENCES

[1] Burrows, S., Gurevych, I., Stein, B. (2015). The eras and trends of automatic short answer grading. International journal of artificial intelligence in education, 25: 60-117. https://doi.org/10.1007/S40593-014-0026-8/TABLES/11

[2] Devlin, J., Chang, M.W., Lee, K., Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805. https://doi.org/10.48550/arxiv.1810.04805

[3] Reimers, N., Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084. https://doi.org/10.48550/arxiv.1908.10084

[4] Ndukwe, I.G., Amadi, C.E., Nkomo, L.M., Daniel, B.K. (2020). Automatic grading system using sentence-BERT network. In Artificial Intelligence in Education: 21st International Conference, AIED 2020, Ifrane, Morocco, pp. 224-227. https://doi.org/10.1007/978-3-030-52240-7_41/TABLES/1

[5] Ahmed, A., Joorabchi, A., Hayes, M.J. (2022). On the application of sentence transformers to automatic short answer grading in blended assessment. In 2022 33rd Irish Signals and Systems Conference (ISSC) Cork, Ireland, pp. 1-6. https://doi.org/10.1109/ISSC55427.2022.9826194

[6] Mohler, M., Bunescu, R., Mihalcea, R. (2011). Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies, pp. 752-762. https://aclanthology.org/P11-1076

[7] Xue, J., Tang, X., Zheng, L. (2021). A hierarchical BERT-based transfer learning approach for multi-dimensional essay scoring. IEEE Access, 9: 125403-125415. https://doi.org/10.1109/ACCESS.2021.3110683

[8] Lubis, F.F., Putri, A., Waskita, D., Sulistyaningtyas, T., Arman, A.A., Rosmansyah, Y. (2021). Automated short-answer grading using semantic similarity based on word embedding. International Journal of Technology, 12(3): 571-581. https://doi.org/10.14716/IJTECH.V12I3.4651

[9] Hasanah, U., Permanasari, A.E., Kusumawardani, S.S., Pribadi, F.S. (2019). A scoring rubric for automatic short answer grading system. Telkomnika (Telecommunication Computing Electronics and Control), 17(2): 763-770. https://doi.org/10.12928/TELKOMNIKA.V17I2.11785

[10] Galhardi, L.B., de Mattos Senefonte, H.C., de Souza, R.C.T., Brancher, J.D. (2018). Exploring distinct features for automatic short answer grading. In Anais do XV Encontro Nacional de Inteligência Artificial e Computacional, pp. 1-12. https://doi.org/10.5753/ENIAC.2018.4399

[11] Hochreiter, S., Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8): 1735-1780. https://doi.org/10.1162/NECO.1997.9.8.1735

[12] Mueller, J., Thyagarajan, A. (2016). Siamese recurrent architectures for learning sentence similarity. In Proceedings of the AAAI conference on artificial intelligence, 30(1): 2786-2792. https://doi.org/10.1609/AAAI.V30I1.10350

[13] Mikolov, T., Chen, K., Corrado, G., Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

[14] Arifin, A.R., Purnamasari, P.D., Ratna, A.A.P. (2021). Automatic essay scoring for Indonesian short answers using siamese Manhattan long short-term memory. In 2021 International Conference on Electrical, Communication, and Computer Engineering (ICECCE), Kuala Lumpur, Malaysia, pp. 1-6. https://doi.org/10.1109/ICECCE52056.2021.9514223

[15] Tulu, C. N., Ozkaya, O., Orhan, U. (2021). Automatic short answer grading with semspace sense vectors and malstm. IEEE Access, 9: 19270-19280. https://doi.org/10.1109/ACCESS.2021.3054346

[16] Roshinta, T.A., Rahutomo, F. (2016). Analisis aspek-aspek ujian esai daring berbahasa indonesia. In Prosiding Sentrinov (Seminar Nasional Terapan Riset Inovatif), 2(1): 645-654.

http://proceeding.sentrinov.org/index.php/sentrinov/article/view/159.
[17] Rahutomo, F., Roshinta, T.A., Rohadi, E., Siradjuddin, I., Ariyanto, R., Setiawan, A., Adhisuwignjo, S. (2018). Open problems in Indonesian automatic essay scoring system. International Journal of Engineering & Technology, 7(4.44): 156. https://doi.org/10.14419/ijet.v7i4.44.26974
[18] Rababah, H., Al-Taani, A.T. (2017). An automated scoring approach for Arabic short answers essay questions. In 2017 8th International Conference on Information Technology (ICIT) Amman, Jordan, pp. 697-702. https://doi.org/10.1109/ICITECH.2017.8079930
[19] Rohde, A., McCracken, M., Worrall, L., Farrell, A., O'Halloran, R., Godecke, E., David, M., Doi, S.A. (2022). Inter-rater reliability, intra-rater reliability and internal consistency of the Brisbane Evidence-Based Language Test. Disability and rehabilitation, 44(4): 637-645. https://doi.org/10.1080/09638288.2020.1776774
[20] Shweta, P., Adhiya, K. (2022). Comparative study of feature engineering for automated short answer grading. In 2022 IEEE World Conference on Applied Intelligence and Computing (AIC) Sonbhadra, India, pp. 594-597. https://doi.org/10.1109/AIC55036.2022.9848851
[21] Wang, Z., Liu, J., Dong, R. (2018). Intelligent auto-grading system. In 2018 5th IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS) Nanjing, China, pp. 430-435. https://doi.org/10.1109/CCIS.2018.8691244
[22] Jong, Y.J., Kim, Y.J., Ri, O.C. (2022). Improving performance of automated essay scoring by using back-translation essays and adjusted scores. Mathematical Problems in Engineering, 2022, Article ID: 6906587. https://doi.org/10.1155/2022/6906587
[23] Rajagede, R.A., Hastuti, R.P. (2021). Stacking neural network models for automatic short answer scoring. In IOP Conference Series: Materials Science and Engineering, 1077(1): 012013. https://doi.org/10.1088/1757-899X/1077/1/012013
[24] Gaddipati, S.K., Nair, D., Plöger, P.G. (2020). Comparative evaluation of pretrained transfer learning models on automatic short answer grading. arXiv preprint arXiv:2009.01303. https://arxiv.org/abs/2009.01303v1
[25] Ghavidel, H.A., Zouaq, A., Desmarais, M.C. (2020). Using BERT and XLNET for the automatic short Answer grading task. In CSEDU, (1): 58-67. https://doi.org/10.5220/0009422400580067
[26] Prabhu, S., Akhila, K., Sanriya, S. (2022). A hybrid approach towards automated essay evaluation based on BERT and feature engineering. In 2022 IEEE 7th International conference for Convergence in Technology (I2CT) Mumbai, India, pp. 1-4. https://doi.org/10.1109/I2CT54291.2022.9824999
[27] Janda, H.K., Pawar, A., Du, S., Mago, V. (2019). Syntactic, semantic and sentiment analysis: The joint effect on automated essay evaluation. IEEE Access, 7: 108486-108503. https://doi.org/10.1109/ACCESS.2019.2933354
[28] Rajagede, R.A. (2021). Improving automatic essay scoring for Indonesian language using simpler model and richer feature. Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control, 6(1): 11-18. https://doi.org/10.22219/KINETIK.V6I1.1196
[29] Thamrin, H., Verdikha, N.A., Triyono, A. (2021). Text classification and similarity algorithms in essay grading. In 2021 4th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI) Yogyakarta, Indonesia, pp. 201-205. https://doi.org/10.1109/ISRITI54043.2021.9702808
[30] Prabhudesai, A., Duong, T.N. (2019). Automatic short answer grading using siamese bidirectional LSTM based regression. In 2019 IEEE international conference on engineering, technology and education (TALE) Yogyakarta, Indonesia, pp. 1-6. https://doi.org/10.1109/TALE48000.2019.9226026