Automatic Short Answer Grading Using a LSTM Based Approach

Udit Kr Chakraborty
Head of Dept. of Computer Science and Engineering
Sikkim Manipal Institute of Technology
Sikkim Manipal University
Sikkim, India
udit.c@smit.smu.edu.in

Anurag Mishra
Dept. of Computer Science and Engineering
Sikkim Manipal Institute of Technology
Sikkim Manipal University
Sikkim, India
anuragmishra.ofc@gmail.com

2023 IEEE World Conference on Applied Intelligence and Computing (AIC) | 979-8-3503-1006-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/AIC57670.2023.10263899

***Abstract*—Short Answer Grading is an emerging application of Natural Language Processing and text processing. Automated Short Answer Grading (ASAG) is the process of evaluating student-written short responses using computer techniques like Machine learning. The ASAG task has been studied for a long time, but because of the difficulties in the research, it still attracts attention. One of the primary limitations of ASAG is the scarcity of domain-relevant training data. The job of ASAG can be approached using a variety of methods, which can be broadly divided between traditional methods using hand-crafted features and methods based on deep learning. Due to the growing popularity of this field, researchers have been using Deep Learning Approaches to address this challenge over the past five years. This paper explores the methods of creating an LSTM model, to test how close this approach will bring the machine score to that of the Human Score.**

***Index Terms*—Long Short-term Memory, Neural Network, Automatic Short Answer Grading, Natural Language Processing**

I. Introduction

Evaluation of learners' responses to assess the learning outcome is perhaps one of the most critical tasks in teaching-learning. Various question types are used as part of formative and summative assessment to assess different aspects of the learning outcome. Some of the other popular types being multiple choice questions, single word answers and essays, short answer type questions play a very significant role in the assessment eco-system. Through this type of question answering, the learners are tested for comprehension, memory, integration of ideas and linguistic abilities simultaneously. However, evaluation of text based answers is a slow and tedious task often relying largely on the intelligence and comprehension of the evaluator. In recent years, considering the widespread growth and fast development of internet and communication technologies (ICT), the reach of education has spread across geographical boundaries resulting in a huge number of registrations, and conventional learning is being fast replaced by e-learning. The advantage of e-learning being anytime, anywhere learning with multiple reruns of recorded sessions, it also demands a fast answer script to transcript outcome. To reduce cost and effort, most e-learning systems have resorted to objective type questions to evaluate learners' responses. Objective type questions come in multiple forms, namely multiple choice, matching options and single word responses, do not require understanding the text. Free text answers or subjective answers which require students to frame sentences integrating multiple ideas and logically build answers, on the other hand, allow students the freedom to express their views and support their ideas while answering the question[1].

The objectivity and easy assessment and quantifiability of multiple choice questions and the other objective question varieties being reasons behind their popularity, these types do not check the learners' knowledge and understanding of the proof and theoretical aspects[2]. Further, a learner during the test may either be in a state of knowledge or of ignorance. Even being in state of knowledge may have several discrete levels like complete or partial knowledge or incomplete or partial state of absence of knowledge or misconception[3].

Close-ended questions not only fail to credit students for partial knowledge but also may credit answers even if the learner is in state of absence of knowledge or partial or full misconception as it is also possible to score in such tests using pure guess work. Evaluating text-based answers of all learners enrolled for a course can be highly time-consuming, expensive, and susceptible to judgment errors arising out of human bias or fatigue. The problem is further complicated due to the manifold increase in enrollments in Universities due to technology-enabled e-learning platforms. Employing multiple evaluators for large numbers of learner responses would increase cost and may return low inter-evaluator score correlation resulting in poor learning experience. The above reasons have necessitated the development of automated methods for the evaluation of text-based answers to facilitate fast, fair, and error-free assessment. Automatic short answer grading (ASAG) is the task of assessing short natural language responses to objective questions using computational methods[4]. The first reference to automatic evaluation of text appeared in 1966 when it was proposed by Page[5]. However, the field didn't see much progress thereafter until the mid-1990s, probably owing to the inherent complexities in natural language processing. The difficulties in text processing occurring out of word sense ambiguities, grammar anomalies and computational overheads still hold major challenges for this task. However, recent

979-8-3503-1006-1/23/$31.00 ©2023 IEEE
DOI: 10.1109/AIC.2023.54

advances in artificial intelligence and computational linguistics have opened up avenues that hold promise for ASAG as they throw up possibilities of grading text responses without having to fully understand the answer[6]. The most common approach to ASAG is to grade a learner's response through comparison with one or more model answers[7].

This paper presents a Deep Learning based technique for evaluating text-based answers. The scores generated are compared with the scores of two human evaluators to assess for accuracy. The dataset, available at kaggle.com[8], consisted of 13000 students' responses, generated by students from standard 7 to 10, along with scores of two human evaluators. The answer length varied from 150 to 550 words. The rest of the paper presents in detail, the description of the model used, the process followed to assess the answers, and discusses the results. A brief section also discusses the available literature relevant to the proposed technique.

II. Literature Survey

This section presents a study of recent advancements in Automated Short Answer Grading (ASAG). While vectorization and vector distance measurement are the approaches used in the majority of studies, additional techniques have also been researched and provided for the sake of comparative analysis. The field of ASAG started with the work of Page[3], and since then, a lot of research has been conducted. However, a universally accepted solution is yet to be found. Burrows [9] released a comprehensive survey that reports on most of the work done before 2015. After 2015, which has been marked as an AI era, most of the work has been done using deep learning or machine learning approaches.

In the works done by Breyer, F. et al. [10], This paper assesses the value of utilizing the E-rater Scoring Engine to check the grade in an article. This research teaches readers about the usage of surface aspects for comparative essay grading, as discussed in previous works such as Burrows, S. et al. [9]. The authors of that paper focused on the evolution of practices in ASAG and mentioned different ASAG methods for the development of short answer grading. However, no universally accepted solution was found from their research.

Galhardi et al. [9] proposed different machine learning approaches for ASAG. The authors used many datasets and applied different ML approaches to these datasets for evaluation. According to the authors, this effort attained 78% accuracy.

also if taking look at The work by Bonthu, S. et al. [11] for ASAG proposed different deep learning based approaches. In this model author did not compare different approaches.

Abdul Salam et al. [12] proposed a deep learning-based methodology for automated Arabic short answer grading. They also mentioned in their paper a hybrid approach that optimizes LSTM (Long Short Term Memory). Their work focuses on different datasets of science subjects in Egyptian schools and returns good results with the limited dataset which was preprocessed for training. Yet, it was still necessary to test the feature extraction procedure. A major limitation of the paper lies in its applicability for Arabic only, and therefore large-scale acceptance cannot be commented upon.

The work of Sarah Hassan et al. [13], which focuses on supervised learning for automated student answer evaluation based on various paragraph embedding approaches, is the first relevant reference to the type of approach chosen for the current research. The authors of this work discussed various deep-learning methods for embedding paragraphs. The authors calculated the overall word vector count for each word in the response and verified its accuracy by contrasting it with other responses. The word vectors were created using paragraph embedding and analyzed using a variety of methods, including regression classifier methods and cosine similarity.

In the works done by Al-Ansari, K. [14] the world gets to know the survey on different word embedding techniques in natural language processing focusing and deep learning algorithms for achieved the best result. For finding word vector from LP different word embedding technique are used. Word embedding techniques are performing different task like machine translation, sentimental analysis, syntactic parsing etc. Word2Vec, continuous bag of word, skip- gram, GloVe, FastText methods are defined for words embedding. In this paper for word embedding few algorithms are used and less datasets are used. From this paper authors get knowledge about following different word embedding techniques.

The technique of The Automated Exam Correction Framework (AECF) for automated short answer grading was proposed by Balaha, H. M[15]. in their paper [15].Using Semantic similarity process finds similarity percentage between two texts. Authors applied different approaches for converting answer data into numeric value. The authors reportedly used RoBERT, the robust BERT model for vectorization of text there by augmenting the popularity of BERT.

For the examination of subjective type answers, Desai, M. B. et al. [16] suggest using Natural Language Processing (NLP) and optical character recognition (OCR) techniques. In this paper different datasets are used. The biggest challenge of this work is to retrieve the data from human written answer sheets with maximumaccuracy [16].Roger Alan Stein et al. (2019) focusing on different machine learning algorithm for text classification. Text classification require transforming the structure text into a standardized numerical representation for easy of analysis.

Word2Vec: Word2Vec was created by Tomas Mikolov[17] and his team at Google in 2013 the main purpose from very large data sets computing continuous word vector representations . The re-searcher and his team were trying to develop faster learning models in a different way from the neural network models that were more simple and popular at that time. Word2vec is a two-layer neural network that processes text via word vectorizing. Word2vec uses text corpora as its input and produces a set of vectors as its output[18].

CBOW: This model, known as a continuous bag-of-words[19], predicts the missing word in a sentence by taking

the words from the sentence and applying them in the projection window to obtain vectors and relationships between the words .

Skip-gram: This variation guesses the closely comparable context terms using just one word as an input. Because of this, it effectively represents uncommon words. GloVe: Global vectors for word representation are known as GloVe. Another word vector representation using the concept of word co-occurrence matrix from a corpus.By creating a co-occurrence matrix for a sizably large corpus, GloVe learns word vectors (word embeddings). These word vectors are taught to capture semantic aspects that are encoded in the ratio between co-occurrence probabilities of words. The cosine distance is used.

BERT (Bidirectional encoder representations from transformers): Google has released BERT as a new technique for obtaining pre-trained word representations from language models. For many NLP jobs, BERT[20] is helpful.This project aims to extract the token embedding from the pre-trained model of BERT. In this method, you can create your model by just using or token embedding rather than creating and fine-tuning an end-to-end NLP model..

FastText: One of the word embedding techniques is FastText. It is an expansion of Word2vec, which was introduced by Facebook in 2016. It is used for text representation and classification. Words are divided into numerous -grams using FastText (sub-word). For example the word vector banana is a sum of the vectors of the n-grams (’ba’, ’ban’, ”bana’, ’banan’, ”banana’, ’banana’, ’ana’. ’nana’. ’anana”nan’, ’an’) this approach will help is helpful for multiple reasons like:Better word embeddings for rare words,Having a vector for those words based on their character n-grams is useful even if the word is not present in the training corpus[21] .

The primary finding which is evident from all the above mentioned research publications is that word embedding and transformer models are popular choices for vectorization of text needed for deep learning based approaches to ASAG. Based on the study, it could be observed that a few popular techniques like BERT and its variants, FastText etc. have been largely used by researchers. However, no comment has been made on the choice of the technique made and its dependence on the text made available, if any.

III. Data Set

The present study utilized a large dataset obtained from the William and Flora Hewlett Foundation competition hosted on kaggle.com[8]. In order to handle the enormity of the dataset, it was split into eight sets of ASCII text essays that were written by students ranging from grades 7 to 10. Each of the eight sets of essays was endowed with unique features that were leveraged to grade them, ensuring that the automated grading system is well-versed in evaluating a diverse range of essay styles. All essays had at least one human score and a null resolved score. Moreover, each essay set had a different grading scheme, ranging from holistic to trait-based. The length of the essays varied from 150 to 550 words, with some writings relying more heavily on their sources than others.

The selection of essays for the dataset was based on stringent data criteria, with the aim of including a variety of essay types to better comprehend the strengths and limitations of the proposed solution. To highlight quality and reliability, the study aimed to match the scores of professional human graders for each essay. The dataset contained multiple scores from human evaluators, which were averaged and compared to the predicted scores obtained using our LSTM model and 5-fold kappa validation[22].

However, the dataset presented certain challenges as a low percentage of essays were awarded the highest scores, which could potentially affect the model’s performance. The distribution of the scores in terms of their percentage is provided below in Fig1 for visualizing the data. Despite these challenges, the proposed solution leveraged the diverse essay styles in the dataset and demonstrated significant accuracy in automated essay grading. Hence LSTM need to be cosidered taking in he requirements of the Data Set.

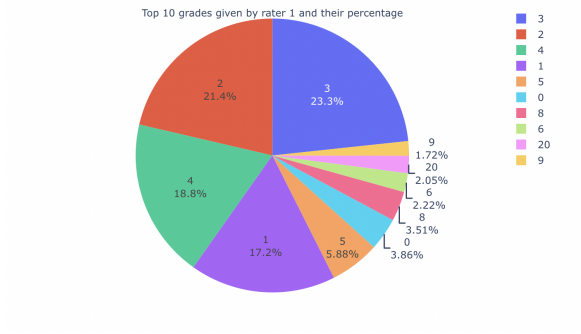

Fig. 1. The distribution of the Scores by their percentage

IV. Why was LSTM Considered

A. LSTM

Long Short-Term Memory Networks (LSTMs) are a specialized class of Recurrent Neural Networks (RNNs) that have the ability to capture long-term dependencies. They were first introduced by Hochreiter and Schmidhuber in 1997 and have since been developed and widely used by various authors[18]. Due to their exceptional performance, LSTMs have become a popular choice for a wide range of applications.

LSTMs are specifically designed to address the issue of long-term dependence. Unlike traditional RNNs, they are not plagued by the vanishing gradient problem and are capable of retaining information over extended periods, making them well-suited for tasks that require memory. This unique property of LSTMs makes them an ideal candidate for our objective and thus was deemed the best approach. The use of LSTMs significantly improves the accuracy of the evaluation process.

All recurrent neural networks are comprised of a series of neural network modules that repeat. In typical RNNs, the

recurrent module is made up of only one Tanh layer, for instance.

B. Methodology and feature extraction

Figure 2 depicts a fundamental block diagram-level implementation process. From each essay, a collection of features was retrieved. This research selected qualities that could serve as proxies for the criteria[23] used by a human grader to evaluate the essay. To learn parameters based on these features, the authors utilized linear regression as their learning model. Scores were predicted for a particular collection of test essays, and these scores were compared to human-graded scores to calculate an error measure.

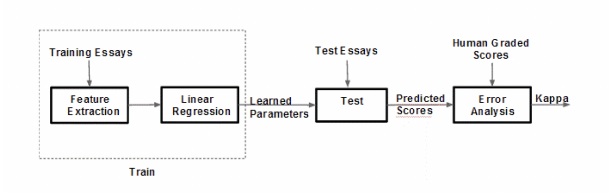

Fig. 2. Implementation Methodology

This paper chose Python 3.7.x as the language to develop its model, given its extensive range of libraries for natural language processing. The authors utilized text mining and the Natural Language Toolkit (NLTK) for most NLP tasks, along with regularized linear regression using Scikit-Learn, besides their own implementation. They also made use of other libraries such as numpy, scipy, xlrd, xlwt, and re for various tasks.

For text preprocessing and feature extraction, the textmining and NLTK packages were used. This mostly involved deleting proper noun placeholders, editing the content, tokenizing, and removing all punctuation from essays. Essay intrinsic variables, such as style and fluency, cannot be quantified directly; instead, they must be estimated by quantifiable factors, such as sentence and word length, essay length, etc. These were the feature extraction from the data set that was done. These features helped train the model.

This paper extracted features across the following categories to train the model: First, Bag of Words (BOW): This was utilized as a starting point to choose characteristics (words) that are reliable indicators of an essay's grade. For making a word bag, the text mining library includes a tool named Term Document Matrix[24]. This is a measurement of the specific content and ideas that the examiner is looking for in each essay set. After deleting stop words like the, of, is, and at from each essay set, the top terms were compiled. They were then compared to the frequency of words used in everyday English. Terms that appear far more frequently than in typical writing were gathered and given weights based on how frequently they were in the essay set. Based on the quantity of these, a value was given to each essay.

Second Number of words per essay, average word length, and number of sentences per essay are all indicators of a writer's language fluency and dexterity. The writings were tokenized and divided using python tools for feature extraction. The numbers for word count, sentence count, and character count were then calculated using each token separately.

Third This paper extracted Different parts of speech, such as nouns and adjectives, count. Verbs and adverbs make suitable stand-ins for vocabulary tests. This characteristic can also be used as a crude stand-in for diction. The NLTK-part-of-speech tagger as found in[25] in Python was used to extract these features. Before the tagging procedure, essays were tokenized into sentences.

Fourth This Research did Orthology Correct word spelling demonstrates a command of the language and ease of use. For this exam, we retrieved the average number of spelling mistakes per essay. The essays were tokenized and punctuation-free. To determine the number of misspelt words per essay, we used the aspell dictionary and the PyEnchant 1.6.5 spell checker.

Using the cosine similarity , paper determine how well each student's response, ai j, matches the required response, ai. To scale the similarities and arrive at the relative measure of similarities, this paper normalise these values from 0 to 1[26]. Authors treat these scores as features of the responses and train them using various regression techniques.

V. Results and Kappa Score

To train and test across each essay set[27], This research made use of the learning model as previously explained. Within each essay set, a 5-fold cross validation was performed. These are the outcomes in table 1 :

Essay Set	Average Kappa
1	0.80
2	0.73
3	0.69
4	0.69
5	0.76
6	0.70
7	0.71
8	0.68
Quadratic Weighted Kappa	0.73

TABLE I
Kappa Values for Essay Sets

Upon getting the predicted values have created a scatter plot to plot the value of the predicted score verses the actual scores of the essays. Then authors converted the predicted values from an NumPy array[28] to a list and extracted them

for further analysis and representation to get the exact image of how good our predictions are.The Results of this model provide a huge insight of how lstm and data set work together. The result show a clear indication of the fact that the data set has lower number of data points on high ranking essays , this is shown in their prediction of the of score when compared to the human score.

VI. Comparison of Predicted vs Actual scores.

The main objective of this paper was to observe how close this model can get the machine score to be to the human score. Hence to achieve that scores of the essay taken as the Test were taken and an average human score was generated . Since the given data set has two human evaluators scoring the essay , we created and average of the scores of the specific essay given by human evaluators. Then the machine predicted score was graphed with the same average score to see the relation and closeness of the two scores.

As it was observed that the machine garneted score generally followed the same patter as that of the human evaluated score. This can be observed that since the data set had low number of high scoring essays as mentioned earlier. The whole machine score is lower in comparison to the human evaluator scores .

The following is the Final plotting of the first 100 essays' predicted score and the average score of the two raters. it would have been hard to compare the predicted score with each moderator so the best way out was to actually extract the score and convert it into an average of the two and then plot it with the predicted scores.

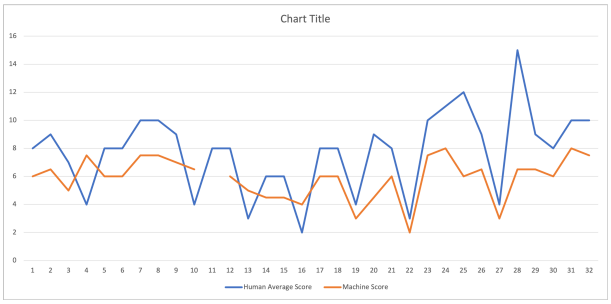

Fig. 3. Predicted VS the Actual values

Here is Fig 3 Authors have the Scores by the model's prediction as in the orange line and the average scores by the human rater given to us in the data set. Authors can clearly observe that the model gives lower scores in to the essays except in few condition because of the lack of high-scoring essays in the data set.Our model works relatively better on noncontext specific essays. Performance on content specific and richer essays can be improved by incorporating content and advanced NLP features.

VII. Conclusion

The LSTM model turns out to be a great fit for the model since the result show that it is able to predict closer to the human score when seen in when comparing it to the human score . The comparison though highlights the fact that the data set lacks or has less of the hight scoring essays in each set . but with the normal average scores the prediction are on the same lines of the human score. the testing results , that are graphed are thus proving the point that the model learns to score the essay in simalar manner the humans do . though there are Multiple outliers presents , to the variable digree Authors can say that in the harware requirement and constrainst this model is trained it has prermosmed quite well on it's own.

The central objective of this research paper was to investigate the extent to which machine-generated scores can approximate human scores. In order to achieve this objective, a set of essays was selected as a test and an average score was generated based on evaluations provided by two human assessors. To create this average score, the individual scores assigned by the two human evaluators were combined. Subsequently, the machine-generated scores were plotted against this average human score in order to assess the degree of closeness and correlation between the two sets of scores. By doing so, Authors sought to determine whether machine-generated scores can be a reliable substitute for human scores in evaluating essays.

For a significant period of time, the task of automatically assessing text-based answers has posed a significant challenge. In order to evaluate student responses and determine the effectiveness of the learning experience, it is necessary to employ a range of question types. Given the widespread use and rapid growth of the internet and associated technologies, the number of registrations for online courses has increased dramatically, highlighting the need for standardized methods of automatically evaluating text-based answers.

Text-based answers can take various forms, including essays, essay-style questions, short answer questions, and single-word answer questions. While automated assessment of single-word answers may be relatively straightforward, complexities arise when evaluating responses that involve sentence construction by the student. Due to the inherent intricacies of word sense disambiguation and semantic evaluation in natural language processing, assessing text-based answers has proven to be a challenging task.Assessing text-based answers is a challenging task due to the inherent intricacies of word sense disambiguation and semantic evaluation in natural language processing.I our own data set we see that the lack of High ranking essays[27] has resulted the machine leaning model to score more stringently.and this has result in most of the machine socres being less than that of Human scores.

Hence this paper has tried to Get a KAppa score of how the LSTM model is performing in training and Testing codnitions and then also tested it on the essays that were not the part of testing or Training data set. This has given authors an accurate measurement and matericts to establish the eficiency of our model. The results are graphed for better representaion and demark the biases toward low score in the model.

REFERENCES

[1] Pintso, M., Doucet, A.V. and Fernandez-Ramos, A. (2010) 'Measuring students-information skills through concept mapping', Journal of Information Science, Vol. 36, No. 4, pp.464–480.

[2] Chang, S.H., Lin, P.C. and Lin, Z.C. (2007) 'Measures of partial knowledge and unexpected responses in multiple-choice tests', Educational Technology and Society, Vol. 10, No. 4, pp.95–109.

[3] Chakraborty, Udit Konar, Debanjan Roy, Samir Choudhury, Sankhayan. (2016). Automatic Short Answer Grading using Rough Concept Clusters. International Journal of Advanced Intelligence Paradigms. 10. 10.1504/IJAIP.2018.10010768.

[4] Immanuel, S. D., amp; Chakraborty, U. Kr. (2019). Genetic algorithm: An approach on optimization. 2019 International Conference on Communication and Electronics Systems (ICCES). https://doi.org/10.1109/icces45898.2019.9002372

[5] Page, E.B. (1966). The imminence of grading essays by computer. Phi Delta Kappan, 47(5), 238–243

[6] Pulman, S.G. and Sukkarieh, J.Z. (2005) 'Automatic short answer marking', Proceedings of the Second Workshop on Building Educational Applications Using NLP, Association for Computational Linguistics, Michigan, pp.9–16

[7] Mohler, M. and Mihalcea, R. (2009) 'Text-to-text semantic similarity for automatic short answer grading', Proceedings of 12th Conference of the European Chapter of the ACL, Athens

[8] The Hewlett Foundation: Short answer scoring. Kaggle. (n.d.). Retrieved April 9, 2023, from https://www.kaggle.com/competitions/asap-sas/data

[9] Burrows, S., Gurevych, I., Stein, B. (2014). The Eras and Trends of Automatic Short Answer Grading. International Journal of Artificial Intelligence in Education, 25(1), 60-117. https://doi.org/10.1007/40593-014-0026-8s, 2014(2), 1-66. https://doi.org/10.1002/ets2.12022T

[10] Breyer, F. J., Attali, Y., Williamson, D. M., Ridolfi-McCulla, L., Ramineni, C., Duchnowski, M., Harris, A. (2014). A Study of the Use of thee-rater® Scoring Engine for the Analytical Writing Measure of the GRE® revised General Test. ETS Research Report Serie

[11] Bonthu, S., Rama Sree, S., Krishna Prasad, M.H.M. (2021). Automated short answer grading using deep learning: A Survey.. In Machine Learning and Knowledge Extraction: Sth IFIP TC 5, TC 12, WG 8.4, WG 8.9, WG 12.9 International Cross-Domain Conference, CD-MAKE 2021, VirtualEvent, August 17-20, 2021,Proceedings5 (pp.61-78).SpringerInternationalPublishing,Cham.

[12] Abdul Salam, M., El-Fatah, M. A., Hassan, N. F. (2022). Automatic grading for Arabic short answer questions using optimized deep learning model. PLOS ONE, 17(8), e0272269. https://doi.org/10.1371/journal.pone.0272269

[13] Hassan, S., A., A., EI-Ramly, M. (2018). Automatic short answer scoring based on paragraph embeddings. International Journal of Advanced Computer Science and Applications, 9(10). https://doi.org/10.14569/ijacsa.2018.091048

[14] Al-Ansari, K. (2020). Survey on Word Embedding Techniques in Natural Language Processing. researchgate.net, August 2020.

[15] Balaha, H. M., amp; Saafan, M. M. (2021). Automatic Exam Correction Framework (AECF) for the mcqs, essays, and equations matching. IEEE Access, 9, 32368–32389. https://doi.org/10.1109/access.2021.3060940

[16] Desai, M. B., Desai, V. D., Gupta, R. S., Mevada, D. D., Mistry, Y. S. (2021). A Survey On Automatic Subjective Answer Evaluation. Advances and Applications in Mathematical Sciences Volume 20, Issue 11, Pages 2749-2765 , Mili Publications, September 2021.

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

[18] Magooda, A. E., Zahran, M., Rashwan, M., Raafat, H., Fayek, M. (2016, March. Vector based techniques for short answer grading. In The twenty-ninth international flairs conference.

[19] Mahana, M., Johns, M., amp; Apte, A. (n.d.). Automated Essay Grading Using Machine Learning - Stanford University. Automated Essay Grading Using Machine Learning. Retrieved February 13, 2023, from http://cs229.stanford.edu/proj2012/MahanaJohnsApte-AutomatedEssayGradingUsingMachineLearning.pdf

[20] Haller, S., Aldea, A., Seifert, C., Strisciuglio, N. (2022). Survey on Automated Short Answer Grading with Deep Learning: From Word Embeddings to Transformers. ArXiv. https://doi.org/10.48550/arXiv.2204.03503

[21] Landauer, T. K., Dumais, S. T. (1997). A Solution to Plato's Problem: The Latent Semantic Analysis Theory of Acquisition, Induction, and Representation of Knowledge. Psychological Review, 104(2), 211-240 DOI: https://doi.org/10.1037/0033-295x.104.2.211

[22] Salim, H. R., De, C. Pratamaputra, N. D. Suhartono, D. (2022). Indonesian automatic short answer grading system. Bulletin of Electrical Engineering and Informatics, 11(3), 1586-1603. DOI: https://doi.org/10.11591/eei.v113.3531

[23] Saha, S., Dhamecha, T. I., Marvaniya, S., Sindhgatta, R., Sengupta, B. (2018). Sentence level or token level features for automatic short answer grading?: Use Both. In: Artificial Intelligence in Education. AIED 2018 London, UK, June 27-30, 2018, Proceedings, Partl 19 (pp. 503-517). Springer International Publishing. DOI: https://doi.org/10.1007/978-3-319-93843-1 37

[24] Lishan Zhang, Yuwei Huang, Xi Yang, Shengquan Yu, Fuzhen Zhuang Year: 2019. An automatic short-answer grading model for semi-open-ended questions. Container: Interactive Learning Environments Page: 1-1 DOI: 10.1080/10494820.2019.1648300 https://doi.org/10.1007/978-3-030-84060-05

[25] Zhu, X. H., Wu, H., Zhang, L. (2022). Automatic Short Answer Grading via BERT-based Deep Neural Networks. IEEE Transactions on Learning Technologies,1-1. https://doi.org/10.1109/tlt.2022.3175537

[26] Stein, R. A., Jaques, P. A., Valiati, J. F. (2019). An analysis of hierarchical text classification using word embeddings. Information Sciences,471, 216-232. https://doi.org/10.1016/j.ins.2018.09.001

[27] The Hewlett Foundation: Short answer scoring. Kaggle. (n.d.). Retrieved April 9, 2023, from https://www.kaggle.com/competitions/asap-sas/data

[28] Galhardi, L. B., Brancher, J. D. (2018). Machine learning approach for automatic short answer grading: A systematic review. In Advances in Artificial Intelligence-IBERAMIA 2018:DOL:https://doi.org/10.1007/978-3-030-03928-8$_3$1