Automatic Essay Scoring: A Review on the Feature Analysis Techniques

Ridha Hussein Chassab, Lailatul Qadri Zakaria, Sabrina Tiun
The Asean Natural Language Processing (ASLAN), Faculty of Information Science and Technology
Universiti Kebangsaan Malaysia, Bangi, Selangor Darul Ehsan, Malaysia

***Abstract*—Automatic Essay Scoring (AES) is the automatic process of identifying scores for a particular essay answer. Such a task has been extensively addressed by the literature where two main learning paradigms have been utilized: Supervised and Unsupervised. Within these paradigms, there is a wide range of feature analyses has been utilized, Morphology, Frequencies, Structure, and semantics. This paper aims at addressing these feature analysis types with their subcomponent and corresponding approaches by introducing a new taxonomy. Consequentially, a review of recent AES studies is being conducted to highlight the utilized techniques and feature analysis. The finding of such a critical analysis showed that the traditional morphological analysis of the essay answer would lack semantic analysis. Whereas, utilizing a semantic knowledge source such as ontology would be restricted to the domain of the essay answer. Similarly, utilizing semantic corpus-based techniques would be impacted by the domain of the essay answer as well. On the other hand, using essay structural features and frequencies alone would be insufficient, but rather as an auxiliary to another semantic analysis technique would bring promising results. The state-of-the-art in AES research concentrated on neural-network-based-embedding techniques. Yet, the major limitations of these techniques are represented as (i) finding an adequate sentence-level embedding when using models such as Word2Vec and Glove, (ii) 'out-of-vocabulary when using models such as Doc2Vec and GSE, and lastly, (iii) 'catastrophic forgetting' when using BERT model.**

Keywords*—Automatic essay scoring; automatic essay grading; semantic analysis; structure analysis; string-based; corpus-based; word embedding

I. Introduction

The last decade has witnessed a dramatic evolution in employing Artificial Intelligence (AI) in the educational domain. This has been represented in classifying questions [32], question answering [10], or question generation [23]. Another challenging area in the educational domain is Automatic Essay Scoring (AES) or Automatic Essay Grading (AEG). AES refers to the task of automatically determining an exact or nearly score for an essay answer [26]. This would require an extensive analysis of the answer's textual characteristics to identify an accurate score. The common method depicted in the literature for doing such an analysis is to acquire a reference answer (sometimes referred to as a model or template answer) and compare it with the student's answer. The comparison would have taken a wide range of forms depending on the technique used for scoring.

Generally speaking, AES's scoring techniques belong to two major categories; Supervised and Unsupervised. According to the machine learning paradigms, such categories refer to the learning mechanism [47]. For instance, in the supervised learning paradigm, a previous or example dataset is being prepared to train the classification algorithm. In this regard, previous students' answers along with reference answers are being arranged along with their actual score given by the teacher and the aim is to train the machine learning algorithm to predict the score of upcoming, testing, or unseen answers. This process of learning is known as regression where the goal is to predict a numeric value rather than a predefined class label (i.e., machine learning classification).

After acquiring the example or training data of answers, a set of numeric features will be generated to predict the score. Such features could be derived from the answer's textual characteristics such as morphology, semantic, structure, or frequency of terms. Regarding the regressor itself, there are a wide range of algorithms have been depicted in the AES literature where it can be categorized into two main classes; traditional regressors and deep learning regressors. Traditional regressors refer to the Linear Regression which is a statistical algorithm that intends to identify the most accurate coefficients that would turn the numeric features of the answers (i.e., X variables) into its actual score (i.e., Y output) [33]. On the other hand, deep learning regressors refer to the latest and sophisticated neural network architectures such as Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN). Such architectures are intended to process the numeric features of the answers through a neural network to predict the output score. Such a prediction is performed through a learning mechanism by randomly generated weights that are linked between the input and output layers within a hidden layer that aims at determining deep relationships [36].

On the other hand, the unsupervised learning paradigm refers to the task of categorizing data without the use of a predefined example or training set, but rather through a distance/similarity function or curated set of rules. The simple and most straightforward mechanism of unsupervised AES is where each answer is compared with reference answer or other student answers for identifying a similarity score which will be used afterward as the final score of the answer. This kind of pairwise similarity can be used separately or through a clustering technique that aims at grouping the similar answers into multiple groups [4].

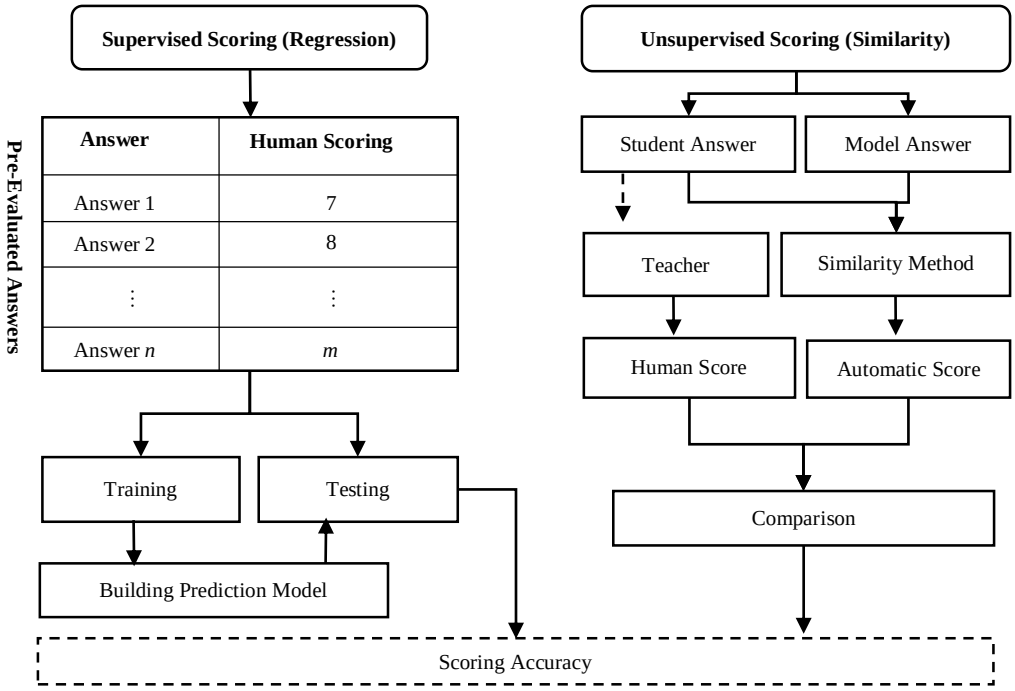

Fig. 1. General Workflow of AES Task.

Another unsupervised technique that has been depicted in the AES literature is the rule-based or ranking approach, the characteristics of the textual answer information in this method are ranked or encoded into a numeric value. Using a predefined set of rules, the numeric values associated with a particular answer would have undergone a summation or averaging procedure to get the overall score. Fig. 1 depicts the general workflow of the AES task.

Based on the general workflow of the AES task in Fig. 1, an extensive literature review is accommodated in this paper where Section 2 will depict such a review. Section 3 will depict the proposed taxonomy where the techniques used by the literature for the AES task are being categorized. Lastly, Section 4 provides a discussion on the techniques where the pros, cons, and ongoing challenges will be determined.

II. Related Work

In this section, an extensive literature review will be conducted on the recent AES research studies. The related works will be divided into two major parts; supervised AES and unsupervised AES. The following subsections will tackle these parts.

A. Supervised AES

In a study of lexical sophistication for evaluating second language writing proficiency (L2), [25] examined two main approaches of lexical techniques. First, the authors have utilized the word frequency where the statistics of the terms within the answers are being exploited. Second, the authors have utilized the n-gram sequences (i.e., bigram) to capture multi-word sequences. Besides, the authors have adopted some ranks for the academic writing and word range. Lastly, a simple regression has been used to predict the score of the answers. Using a corpus for the English placement test (i.e., TOFEL), the proposed method showed 92.6% of accuracy.

The author in [15] has treated the AES task as a regression problem where a Support Vector Regressor (SVR) has been used to predict the score of Portuguese student answers. For this purpose, a dataset obtained from Brazilian Schools has been used. In addition, the input of the SVR was represented as a set of numeric features that have been obtained by the structure of the answer, lexical diversity, theme, and coherence. The authors have defined scores for each feature and then use them as input to the SVR. The authors reported 74.7% as a value of correlation between regression result and teacher score. However, this study has used an imbalance dataset where five scores are used as 0, 50, 100, 150, and 200. The classes of the students' answers were imbalanced therefore, latterly the authors have proposed an improvement in their work of [14]. Using some statistical algorithms, the authors have managed to improve the accuracy of assessment by obtaining 75% of correlation.

The author in [11] has treated the AES task as regression problem where SVR algorithm has been utilized. In order to input the answer text to the regressor, the authors have used a method called Bag-of-Super-Word-Embeddings (BOSWE). This method works on existing embedding vector of words to accommodate clustering where the centroid terms will be represented as super words. Hence, the semantic meaning of terms would be converted into group of clusters. Consequentially, the authors have utilized a string similarity called Histogram Intersection String Kernel (HISK) which is a measure that has been widely used to calculate similarity between images' histograms. In this regard, the histogram of word embedding clusters would be targeted. To obtain the word embedding vectors, the authors have used a pre-trained

model based on Word2Vec introduced by [31]. An English benchmark dataset of Automated Student Assessment Prize (ASAP) has been used where the accuracy result was 78.8%.

The author in [19] established a comparison between pre-trained word embedding models and paragraph embedding models. For the pre-trained word embedding, the authors used Google Word2Vec, Glove, FastText and Elmo models. Whereas, for the paragraph embedding, the authors used Doc2Vec, InferSent and SkipThought models. Lastly, cosine similarity used to determine the similarity between student's answer vector and teacher's answer vector. Such similarity will be fed into a Ridge regression classifier. Using a benchmark dataset of English questions and answers brought from University of North Texas, the authors have concluded that the paragraph embedding using Doc2Vec has achieved the highest correlation of 56.9%.

The author in [27] proposed a self-attention method that captures long-distancer relationship for AES task. The authors first utilize the self-attention network to process two inputs including word vector embedding and word position. The word embedding has been brought from a pre-trained model of Glove where the average of each word's vector within a sentence is gained through padding approach. Another input will be depicted for the word position where each word would have a position embedding. The output of the self-attention will be processed via a Long Short-Term Memory (LSTM) architecture to accommodate the scoring. Using the benchmark of ASAP, the proposed method showed an accuracy of 77.6%.

The author in [45] proposed a deep learning architecture for the AES task. The proposed architecture begins with Word2Vec embedding for the words within the student's answer. Consequentially, the resulted embedding will be processed via a bidirectional LSTM in order to extract semantic features. Lastly, an attention layer will process the extracted features in order to give the score. The benchmark dataset of ASAP has been used in the experiments where the acquired accuracy was 83%.

The author in [41] has proposed a hybrid method of Support Vector Machine (SVM) and LSA to provide automated assessment of answers in Indonesian languages. The authors used a dataset contains students answers along with lecturer answers. Then, they treated the problem as topic modeling where SVM has been used to classify the answers into multiple generated number of topics. If a particular student answer has been classified into an irrelevant topic in respect to the lecturer answer, it would be assessed as zero.

The author in [17] examined the lexical sophistication for evaluating second language writing proficiency (L2) where Korean students are being tested on English placement test. The authors have treated the problem as regression in which the n-gram features of multi-word sequences are being addressed. For this purpose, the authors prepared answers from native speakers and compare it with the tested answers. Within such a comparison the authors addressed the occurrence of bigram and trigram sequences. A corpus of English placement test has been used to accommodate the comparison. The comparison aims at computing the associate measure between n-gram sequences. Lastly, the statistical measures' values will be fed to a step-wise regression in order to predict the automatic score. Results of correlation between automatic and human score were 84.64%.

The author in [9] has treated the task of AES differently where the problem has been handled as a regression task. Instead of accommodating feature engineering on the answer text, the authors have used the answer as an input to a Convolutional Neural Network (CNN) architecture that has been incorporated with a regression layer. Such regression layer will predict the score of the answer based on a non-linearized function. To do so, the authors have input the embedding of words inside the answer to the architecture. Such embedding has been obtained via a pre-trained Glove model proposed by [48]. An English benchmark dataset of Automated Student Assessment Prize (ASAP) has been used. Results of accuracy obtained were 82.6%.

The author in [28] proposed a multi-way attention architecture for AES task. The proposed architecture contains a transformer layer at first which process pre-trained Glove word embedding of student's answer and model's answer. Then, the following layer represents the multi-way attention where three self-attention vectors are represented for the student's answer, model's answer and their cross vector respectively. This will be followed with an aggregation layer where word's position vectors will be added. The final layer contains the regressor where the score of the essay is being predicted. For this purpose, the authors have used a real-word educational dataset of questions and answers. Result of accuracy was 88.9%.

The author in [46] proposed a deep learning architecture for AES task. The proposed architecture begins with pre-trained word embedding vectors brought from Glove and processed via CNN layer. Then, the resulted features will be processed via LSTM in order to generate sentence embedding for each answer. The key distinguishes of this study lies in adding a co-attention layer that consider the similar sentences between student's answer and model's answer. Lastly, the final layer will give the score for each answer. Using ASAP benchmark dataset, the proposed architecture produces an accuracy of 81.5%.

The author in [34] has examined the possibility of incorporating embedding features with structural features or so-called feature-engineered. The authors have utilized an LSTM where sentence-level embedding incorporated with a set of feature-engineered. Using ASAP dataset, the proposed method showed an accuracy of 77.5%.

The author in [24] examined the lexical sophistication for evaluating second language writing proficiency (L2). The authors have used a corpus for English placement test (i.e., TOFEL). Using some lexical features such as word and n-gram overlapping along with a semantic approach of LSA, the authors have applied a simple regression in order to predict the score of the tested answers.

The author in [26] has proposed a deep learning method for AES task where two architectures of CNN and LSTM are being employed. First, the authors have processed the words' vectors of each answer through the CNN architecture in order to get the sentence embedding. For this purpose, a pre-trained

model of Glove word embedding has been used. In addition, the resulted sentence embedding from CNN has been furtherly processed via the LSTM architecture in order to get the score. Using the benchmark dataset of ASAP, the authors have shown an accuracy of 72.65%.

The author in [44] has proposed a deep learning architecture for AES task. The proposed architecture begins with word embedding vectors generated by Word2Vec and process via CNN layer in order to extract n-gram features. Lastly, a recurrent layer called Bidirectional Gated Recurrent Unit (BGRU) is being used to predict the score of the answer. Using the benchmark dataset of ASAP, the proposed architecture showed an accuracy of 86.5%.

The advancement of deep learning architecture led to the emergence of Transformers which yield a novel mechanism in learning. Such mechanism lies in the synchronized bidirectional learning. Such an architecture led to the emergence of Bidirectional Encoder Representations from Transformers (BERT) embedding. BERT has a fixed and indexed pretrained model of embedding where a vocabulary of 30,000 English terms is being stored. BERT has shown remarkable superior performance in text generation applications.

However, recently, [43] have utilized the BERT architecture for the AES task. Using ASAP dataset, BERT showed an accuracy of 74.75%. The authors have compared the BERT against the LSTM and the comparison showed that LSTM is still a competitor where it achieved an accuracy of 74.63%. The authors have justified such a miscarriage of BERT regarding a problem known as 'catastrophic forgotten' where the BERT architecture would forget quickly what it had learnt previously.

Similarly, [30] has proposed a BERT architecture for the AES task. The authors have utilized the pretrained BERT embedding and then apply the fine-tune. Using ASAP dataset, results of accuracy showed an average of 64.6% achieved by the proposed BERT.

The author in [42] has examined a Multi-Task Learning (MTL) of AES where the essay is being assessed as traits rather than holistic (i.e., Single-Task Learning). The authors have utilized structural features as traits such as the organization of the essay, the discourse of topic, and the vocabulary size of the essay; in addition, a word embedding CNN architecture using through Glove along with a sentence embedding through LSTM. The traits have been encoded through a pooling attention layer. Using ASAP dataset, the proposed MTL showed an accuracy of 76.4%.

The author in [35] has utilized much more efficient architectures derived from BERT such as Albert and Reformer for the AES task. In fact, BERT suffers from the tremendous extent of parameters (around 60 million). Therefore, the authors have concentrated on architectures that derived from BERT with considerably lower number of parameters. Using the ASAP dataset, the authors have demonstrated that the proposed architectures maintained fair accuracy of 78.2% with significant drop in the computational requirements.

B. Unsupervised AES

The author in [18] has introduced the first benchmark of Arabic dataset for automatic scoring essays which contains 610 students' answers written in Arabic language. The domain of question was geography. The authors have applied several similarity measures including string-based, n-gram and corpus-based specifically Distributional Semantic Co-occurrence (DISCO) similarity measures independently and with combination. Then they have applied k-means clustering approach in order to scale the obtained similarity values. Results of correlation between manual and automatic score were 83%.

The author in [37] has established a comparative study on two main similarity approaches through an unsupervised paradigm for AES task. The authors have firstly used the Cosine measure to compute the similarity between student's answer and model's answer. Then, the authors have used a corpus-based method of LSA to compute the similarity between the two answers. Using a real-word dataset of questions and answers, LSA showed better performance by obtaining a correlation of 59.7%.

The author in [22] has proposed a ranking algorithm for Automatic Essay Scoring (AES) based on structural and semantic features. The structural features included number of words, number of verbs, number of sentences, and number of paragraphs in an essay, whereas semantic features brought from a corpus-based approach known as Kullback - Leibler divergence. An English benchmark dataset of Automated Student Assessment Prize (ASAP) has been used.

The author in [1] proposed an automatic essay grading system that has been utilizing ontology-based approach. The proposed approach aimed at focusing on the subject of the answer given by the students. For this purpose, the WordNet ontology has been exploited which can provide domain-specific semantic correspondences. The authors have prepared a teacher guide answer in order to be used as a benchmark when evaluating student's answer. Comparing both the guide answer and the student's answer through querying the included terms over WordNet, the authors have computed the similarity using semantic relatedness measure known as Least Common Sub-sumer (LCS). Results of the comparison were set as the automatic score where the Pearson metric has been used to compute the correlation between the automatic score and the teacher score. Experimental results showed an average correlation of 80% has been achieved.

The author in [2] has proposed a system for Arabic AES for a Saudi Intermediate school children. The criteria used for assessing the students' answers were based on spelling, grammar, structure of the answer, relation of the answer to the desired topic, and following the Modern Standard Arabic (MSA) words. For evaluating a particular answer, the authors have adopted a hybrid method of LSA RST. LSA was intended to measure the semantic similarity of the tested answer while, RST was intended to measure the cohesion and the writing style of the answer. The authors have collected a set of pre-evaluated answers of 300 essays where such answers have been written by the intermediate level students from different topics with a score out of 10. Consequentially, the authors have re-

typed the answers to the computer in order to use them for training and testing purposes. Using Pearson Correlation, the authors have compared the automatic score to the teacher score. Experimental result showed an average of 78.3% of Pearson Correlation.

The author in [39] has proposed a fingerprinting method for automatic essay scoring in Japanese language. The authors have utilized a hashing method for each essay answer by computing the ASCI values of the characters within the answer's words. This would provide a distinctive fingerprint for every answer. Then, using a model answer, answers that have been assessed with full mark, the fingerprints of the students' answers will be compared to the fingerprint of the model answer. Since the fingerprint values are numeric thus, Cosine similarity has been used to compute the distance. Based on a set of pre-assessed answers by human (i.e., teachers), the automatic score has been compared to the human's score in order to calculate the accuracy. Lastly, the authors have manipulated some parameters of fingerprint calculation such as number of N-gram characters to get the best accuracy. The proposed method managed to obtain 86.86%.

In another study, [12] described a system called TAALES which has been proposed for the AES task. The proposed system utilizes traditional features to evaluate student answer such as word frequency, academic language, N-gram frequency and other structural features. The proposed system works on user generated text.

The author in [19] has focused on the preprocessing tasks utilized for the AES task in the Indonesian language. The authors have used a corpus of questions, students' answers and teacher answers written in Indonesian. Five preprocessing tasks have been applied including lower-case, tokenization, punctuation removal, stopword removal and stemming. Lastly, Cosine similarity has been used to compute the distance between teacher's answer and student's answer. Results of correlation between manual and automatic scoring were 47%.

The author in [38] has extended the fingerprinting algorithm presented in (A. Agung Putri Ratna et al., 2018) by adding a semantic similarity of LSA. Using a real-word of Japanese questions and answers, the proposed method obtained 87.78% of accuracy. At the same year, the authors have also presented another study (A. A. P. Ratna, Noviaindriani, Santiar, Ibrahim, & Purnamasari, 2019) where the authors have addressed the use of k-means clustering with LSA for AES in Japanese language. Using the same dataset, the proposed method showed an 89% of accuracy.

The author in [38] has proposed an LSA method to provide automated assessment of answers in Indonesian languages. The authors used a dataset contains students answers along with lecturer answers. Then, LSA has been used to calculate the similarity between the two answers. Using a comparison between the automatic score and human score, the proposed method showed an accuracy of 72.01%.

The author in [3] has proposed a rule-based system for Arabic AES that is evaluating the answers based on style issues such as spelling, structure, and coherence. The proposed system utilizes a predefined set of rules that check each essay answer in terms of the aforementioned style aspect. The authors have used online dialogues and discussions among university students in order to train their system. Results of correlation between the automatic and human grading were 73% (Unsupervised).

The author in [5] has proposed an AES system that utilizes LSA along with RST. The authors have built a dataset from the scratch where a set of religious questions has been initiated along with their model answers. Then, such questions have been given to school students in order to answer them. Lastly, a set of teachers has been assigned to give a human / manual scoring that will be used later for training both the LSA and RST. Experimental result showed a 75.6% of correlation between the proposed method's scoring and the teacher's scoring.

The author in [21] has established a comparison among different embedding approaches for the AES task. The authors have utilized the benchmark dataset of ASAP. In addition, traditional vector representations such as TFIDF and Jaccard have been utilized. Furthermore, different embedding approaches such as Glove, Elmo, Google Sentence Encoder (GSE) have been also used. Lastly, using cosine similarity, the authors have identified the similarity between vectors of student answers and vectors of teacher answers. Results showed that the highest correlation achieved by GSE where it obtained 74.3%.

III. Taxonomy of AES Feature Analysis

Within the textual analysis of answers by either the supervised or unsupervised techniques, there is a wide range of features that could be used. Based on the review of literature in the previous section, this section attempts to provide a taxonomy of the AES feature analysis. As shown in Fig. 2, the taxonomy of AES's techniques is divided through the two topologies of supervised and unsupervised paradigms. However, the key characteristics of utilizing the two paradigms lies on the type of feature analysis. In fact, there are four main categories of feature analysis: Structure, Frequency and Term Occurrence, Morphology, and Semantic. Following subsections will tackle each category independently.

A. Structure

In this type of feature analysis, the essay answer is analyzed in terms of its structure where the coherence, writing style and spelling mistakes are being considered. The common approach for this type of feature analysis is the Rhetorical Structure Theory (RST) (Al-Jouie & Azmi, 2017). RST is a linguistic method that aims at analyzing parts of text in order to identify relations among them; it has been widely used for text summarization.

Usually, this type of feature analysis is exploited by the unsupervised technique through a ranking procedure that gives score for each criterion [22], or it could be exploited within a set of rules ([2]; [5]; [12]). On the other hand, this feature analysis can be utilized by a supervised technique through the ranking procedure where the numeric ranks would be fed to a regressor ([14]; [15]).

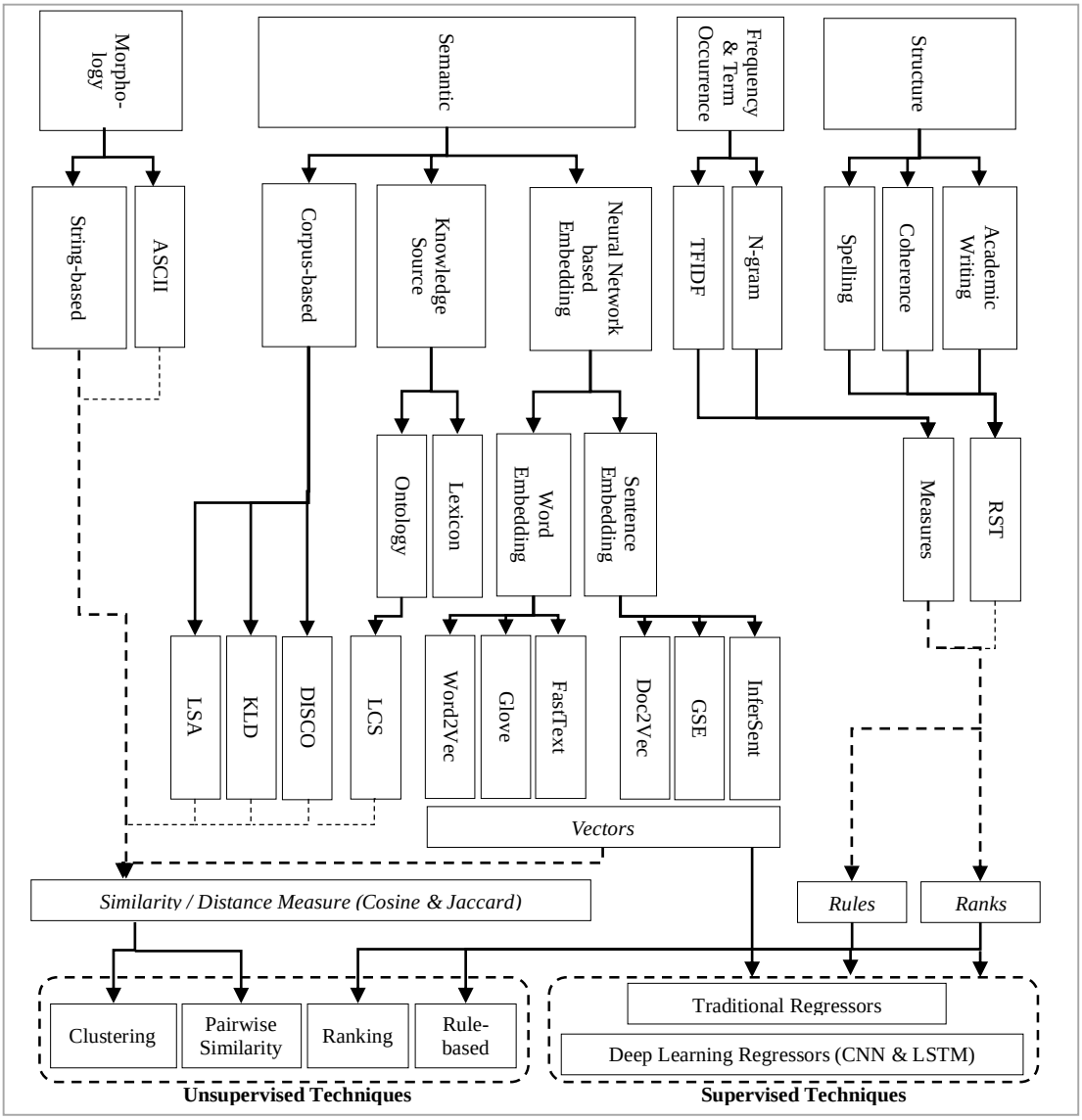

Fig. 2. Taxonomy of AES Feature Analysis.

B. Frequency and Term Occurrence

This type of feature analysis aims at analyzing the frequencies of specific terms or the number of words, sentences and paragraphs. The common approach for counting word frequencies is the Term Frequency Inverse Document Frequency (TFIDF). On the other hand, this feature analysis focuses on the occurrence of specific term and its surrounding words to assess the student's answer. The common approach for this occurrence analysis is the N-gram where the occurrence of unigram (i.e., single term), bigram (i.e., two terms), and trigram (i.e., three terms) can be considered.

The way of utilizing such feature analysis by an unsupervised technique is represented by a set of rules that determine the consequences of capturing specific frequencies or occurrences [12]. Otherwise, the statistics of frequency and occurrence could be exploited directly by a supervised regression technique [25].

C. Morphology

This type of feature analysis concentrates on the lexical morphology of terms within the essay answer. The most straightforward example of this analysis is the string-based similarity between words which can be identified through similarity measures such as Cosine and Jaccard. In addition, sometimes the morphology of words could be extended to consider the ASCII code representation of characters within the essay answer ([38]; [39]; [40]). The way of adopting this feature analysis into an unsupervised technique is simply represented through the use of clustering where similarity values between answers (produced by Cosine or Jaccard) are being used to aggregate similar answers in a single cluster [18]. Otherwise, it could be adopted through a supervised regression technique by processing the similarity values and predicting the score [19].

D. Semantic

This type of feature is considered to be much more sophisticated where the semantic meaning of the answer's words is being analyzed. Apparently, the common way to utilize the semantic aspect is to utilize an external knowledge source such as a dictionary. However, there are other two techniques that can analyze the semantic without using knowledge source depicted in the literature; corpus-based and neural-network-based embedding. The three aforementioned techniques will be illustrated in the following.

1) Knowledge source: In this technique a lexicon, dictionary or ontology is being used to clarify semantic correspondences. Using a knowledge source would offer different semantic relationship between the words such as hypernymy and synonymy which might enhance the comparison between the student answer and the reference answer. WordNet is the most popular ontology that has been used for this purpose. Usually, this feature is utilized by an unsupervised technique through semantic relatedness measures such as Least Common Subsumer (LCS) [1].

2) Corpus-based: In this technique the semantic is being analyzed statistically and without utilizing any knowledge source. In fact, this technique aims at exploiting a corpus of text in order to identify similar contexts which usually yield semantically matching terms. To do that, a matrix of the terms along with their corresponding documents is being initiated. Consequentially, some dimensionality reduction such as Singular Value Decomposition (SVD) is applied to determine semantic correspondences. The most popular corpus-based approaches are the Latent Semantic Analysis (LSA), Latent Dirichlet Allocation (LDA), Distributional Semantic Co-occurrence (DISCO), and Kullback - Leibler Divergence (KLD). The utilization of these approaches for supervised AES is simply represented by feeding a regression technique with answer document vectors [38]. Otherwise, a distance measure such as Cosine can be used to determine similarity between answers documents through an unsupervised technique ([2]; [6]; [18]; [40]).

3) Neural-Network-based-embedding: Similar to the corpus-based approaches, this technique aims at analyzing the semantic aspect of the text without the use of knowledge source. The key distinguishes here is the utilization of neural network architectures to produce special embedding. The earliest effort of this technique was represented by generating distinctive embedding vector for words which referred to as Word Embedding. The most common architectures of word embedding are Word2Vec [45], Glove [9], and FastText [19]. Afterwards, other textual levels have been examined in terms of neural network embedding such as Document Embedding, Sentence Embedding, and Paragraph Embedding. The common architectures for these levels are Doc2Vec, Google Sentence Encoder (GSE), and InferSent [19]. The way of utilizing word and sentence embedding by an unsupervised technique is simply through vector similarity computed by either Cosine or Jaccard. Otherwise, the embedding vectors would be fed to a supervised regression technique in order to predict the score.

Recent years reveal a new embedding architecture of BERT which is based on transformer learning. Such an architecture has the ability to overcome the 'out-of-vocabulary' problem. In addition, it has the capability to handle word-level and sentence-level embedding. The common way of utilizing BERT architecture for the AES task is through a supervised learning ([30]: [43]).

E. AES Datasets

The literature depicts a diversity in using various types of datasets for the AES task. First, there were several languages depicted by the literature such as English, Arabic, Indonesian, Japanese, Portuguese, and others. On the other hand, some efforts utilized synthesis data where students are tested to collect their results. Other efforts utilized real-world data where students' answers from previous exams have been collected. Lastly, the rest of the studies concentrated on benchmark datasets. There are two main benchmark datasets for AES, namely, Automated Student Assessment Prize (ASAP) [29] which has been presented in Kaggle.com as a challenge, and the second dataset is the ETS Corpus of Non-Native Written English from the Linguistic Data Consortium (LDC) [8].

F. Summary of Related Work

To conclude all the researches in AES, Table I shows the summary that briefly describes each related work, whereas, Table II depicts the summary of techniques used by the literature.

TABLE I. SUMMARY OF RELATED WORK

Author	Learning Paradigm	Method	Features	Dataset & Language	Accuracy	Limitations
Gomaa & Fahmy (2014)	Unsupervised	K-means clustering	string-based and corpus-based (DISCO)	Benchmark Arabic dataset of questions and answers	83%	String-based similarity suffers from ignoring semantic aspect. Whereas, corpus-based similarity of DISCO suffers from domain dependent
Kyle and Crossley (2016)	Supervised	Simple regression	Word frequency, n-gram, and academic writing	Real-world student answers (English Placement Test)	92.6%	More semantic features are needed
Pramukantoro and Fauzi (2016)	Unsupervised	LSA + Cosine	String-similarity + semantic similarity	Real-world student answers (English)	59.7%	corpus-based similarity of LSA suffers from domain dependent

Kopparapu and De (2016)	Unsupervised	Ranking algorithm	Structural features (# words, sentences and paragraphs) + Semantic corpus-based (Kullback - Leibler divergence)	English Benchmark Automated Student Assessment Prize (ASAP)	-	corpus-based similarity of Kullback - Leibler divergence suffers from domain dependent
Ajetunmobi and Daramola (2017)	Unsupervised	Ontology-based approach (WordNet)	Semantic relatedness (LCS)	Synthesis (English)	80%	ontology offers domain-specific semantic correspondences where open domain answers would not be assessed effectively
Al-Jouie and Azmi (2017)	Unsupervised	LSA + RST	Semantic similarity + Spelling + structure + grammar	Real-world student answers (Arabic)	78.3%	corpus-based similarity of LSA suffers from domain dependent
Crossley and Kyle (2018)	Unsupervised	Rule-based	Word frequency, n-gram frequency and academic writing	Synthesis (based on user-generated data)	-	more semantic analysis is needed
Hasanah et al. (2018)	Unsupervised	Cosine Similarity	Lowercasing, tokenization, punctuation removal, stopword removal and stemming	Real-world student answers (Indonesian)	47%	more semantic analysis is needed
Ratna et al. (2018)	Unsupervised	Fingerprinting algorithm	Characters ASCI values of the answer's words	Real-world student answers (Japanese)	86.86%	Focusing on ASCI values of characters would only determine morphological similarity and ignore the semantic similarity
Filho et al. (2018)	Supervised	SVR	Predefined numeric scores of features (Structure+ lexical diversity + theme + coherence)	Real-world student answers (Portuguese)	74.7%	More semantic analysis is needed. In addition, much more sophisticated regressor is needed (SVR is considered shallow neural network and not deep learning)
Cozma et al. (2018)	Supervised	Bag-of-Super-Word-Embeddings (BOSWE) + SVR	Histogram Intersection String Kernel (HISK)	English Benchmark Automated Student Assessment Prize (ASAP)	78.8%	This method could suffer from 'out-of-vocabulary' problem. In addition, much more sophisticated regressor is needed (SVR is considered shallow neural network and deep learning)
Hassan et al. (2018)	Supervised	Ridge regression based on vector cosine similarity	Word embedding models (Word2Vec, FastText, Glove, Elmo) + Paragraph embedding models (Doc2Vec, InferSent, Skipthought)	English Benchmark dataset (University of North Texas)	56.9%	Much more sophisticated regression is needed. In addition, regression can benefit from embedding vector features rather than feeding only on similarity values produced by cosine
Li et al. (2018)	Supervised	LSTM regression	Self-attention with Glove embedding and word position	ASAP	77.6%	Glove embedding suffers from 'out-of-vocabulary' problem.
Wang et al. (2018)	Supervised	Bidirectional LSTM with attention layer	Word2Vec word embedding	ASAP	83%	Word2Vec embedding suffers from 'out-of-vocabulary' problem.
Ratna et al. (2019)	Unsupervised	Fingerprinting algorithm + LSA	Characters ASCI values of the answer's words + semantic similarity	Real-world student answers (Japanese)	87.78%	corpus-based similarity of LSA suffers from domain dependent
Ratna et al. (2019)	Unsupervised	K-means clustering + LSA	Semantic similarity	Real-world student answers (Japanese)	89%	corpus-based similarity of LSA suffers from domain dependent
Garner et al. (2019)	Supervised	Stepwise regression	Association measures of bigram and trigram	Real-world student answers (English Placement Test)	84.64%	More semantic analysis is needed. In addition, much more sophisticated regressor is needed

Filho et al. (2019)	Supervised	SVR to solve imbalance classes	Predefined numeric scores of features (Structure+ lexical diversity + theme + coherence)	Real-world student answers (Portuguese)	75%	More semantic analysis is needed. In addition, much more sophisticated regressor is needed (SVR is considered shallow neural network and not deep learning)
Ratna et al. (2019)	Combination: Supervised (topic-modeling) Unsupervised (answer similarity)	SVM + LSA	Topic modeling (SVM) + semantic similarity (LSA)	Real-world student answers (Indonesian)	72.01%	corpus-based similarity of LSA suffers from domain dependent
Nadeem et al. (2019)	Supervised	LSTM	Doc2Vec + Structural Features	ASAP	77.5%	Doc2Vec embedding suffers from 'out-of-vocabulary' problem.
Alqahtani and Alsaif (2019)	Unsupervised	Rule-based	Spelling + structure + coherence	Real-world student answers (Arabic)	73%	More semantic analysis is needed. In addition, the dataset used was relatively small and note adequately adjusted
Azmi et al. (2019)	Unsupervised	LSA + RST	Semantic similarity + Spelling + structure + grammar	Real-world student answers (Arabic)	75.6%	corpus-based similarity of LSA suffers from domain dependent
Chen and Zhou (2019)	Supervised	CNN + Ordinal Regression	Pre-trained word embedding based on Glove	ASAP	82.6%	Glove embedding suffers from 'out-of-vocabulary' problem.
Liu et al. (2019)	Supervised	Multi-way attention architecture	Pre-trained Glove word embedding	Real-world student answers (English)	88.9%	Glove embedding suffers from 'out-of-vocabulary' problem.
Zhang & Litman (2019)	Supervised	LSTM with co-attention layer	Glove pre-trained embedding	ASAP	81.5%	Glove embedding suffers from 'out-of-vocabulary' problem.
Rodriguez et al. (2019)	Supervised	BERT architecture	BERT pretraining embedding	ASAP	74.75%	BERT architecture suffers from 'catastrophic forgetting' problem
Hendre et al. (2020)	Unsupervised	vector embedding cosine similarity	TFIDF, Jaccard, Glove, Elmo, GSE-lite, GSE-large	ASAP	74.3%	Relying on cosine to compute similarity between vectors would seem insufficient, utilizing the embedding features of GSE for a regression task can be seen promising
Kyle (2020)	Supervised	Simple regression	Word frequency, n-gram and LSA	Real-world student answers (English Placement Test)	-	corpus-based similarity of LSA suffers from domain dependent
Li et al. (2020)	Supervised	LSTM regression	Sentence embedding using CNN based on Glove word embedding	ASAP	72.65%	Glove embedding suffers from 'out-of-vocabulary' problem.
Tashu (2020)	Supervised	BGRU	Word embedding Word2Vec processed via CNN	ASAP	86.5%	Word2Vec embedding suffers from 'out-of-vocabulary' problem.
Mayfield and Black (2020)	Supervised	BERT architecture	Pretraining BERT embedding	ASAP	64.6%	BERT architecture suffers from 'catastrophic forgetting' problem
Ridley et al. (2021)	Supervised	Bidirectional LSTM with attention layer	Glove word embedding and LSTM sentence embedding with traits attention layer	ASAP	76.4%	Glove embedding suffers from 'out-of-vocabulary' problem.
Ormerod et al. (2021)	Supervised	Efficient BERT architecture	Efficient architectures derived from BERT such as Albert and Reformer	ASAP	78.2%	Still suffers of 'catastrophic forgetting' problem

TABLE II. SUMMARY OF TECHNIQUES

Author	Morphology		Corpus-based			Embedding					Knowledge	Frequencies & Structural
	String	*ASCII*	*LSA*	*DISCO*	*KLD*	*Word2Vec*	*Glove*	*Doc2Vec*	*GSE*	*BERT*	*Ontology*	
Gomaa & Fahmy (2014)				√								
Kyle and Crossley (2016)												√
Pramukantoro and Fauzi (2016)	√		√									
Kopparapu and De (2016)					√							√
Ajetunmobi and Daramola (2017)											√	
Al-Jouie and Azmi (2017)			√									√
Crossley and Kyle (2018)												√
Hasanah et al. (2018)	√											
Filho et al. (2018)												√
Cozma et al. (2018)						√						
Hassan et al. (2018)						√	√	√				
Li et al. (2018)							√					
Wang et al. (2018)						√						
Ratna et al. (2019)		√	√									
Ratna et al. (2019)			√									
Garner et al. (2019)												√
Filho et al. (2019)												√
Ratna et al. (2019)			√									
Nadeem et al. (2019)								√				√
Alqahtani and Alsaif (2019)												√
Azmi et al. (2019)			√									√
Chen and Zhou (2019)							√					
Liu et al. (2019)							√					
Zhang & Litman (2019)							√					
Rodriguez et al. (2019)										√		
Hendre et al. (2020)							√		√			√
Kyle (2020)			√									√
Li et al. (2020)							√					
Tashu (2020)						√						
Mayfield and Black (2020)										√		
Ridley et al. (2021)							√	√				√
Ormerod et al. (2021)										√		

IV. Discussion

AES task has been depicted in the literature through various techniques. The traditional ones were concentrating on the essay structure, spelling and grammatical errors ([2]; [3]; [6]; 12; [22]). Obviously, these techniques are focusing on general features of the essay and cannot provide an accurate scoring based on such general features. However, the structural features showed feasibility when combined with other semantic features.

Another type of features depicted in the literature is the ones focused on morphological aspect of the words within the essay. This has been represented by utilizing the string-based similarity ([18]; [39]; [40]). On the other hand, the statistics of words within the essay have been also utilized ([12]; [21]). The main limitation behind the aforementioned techniques lies in the absence of semantic analysis in which focusing only on the morphology of the words would discard the semantic factor.

For this purpose, some studies have utilized a semantic knowledge source in order to enhance semantic analysis. For instance, [1] have utilized the ontology of WordNet for AES task. Yet, the problem of such external knowledge source is that it can provide general synonyms of the words where the aim sometime is to focus on specific domain. On other hand, some studies attempted to include the semantic analysis without the use of any external knowledge source, but rather through corpus-based approaches ([2]; [6]; [18]; [40]). These approaches are being fed with specific corpus to analyze the similar contexts which reflects on finding semantic correspondences. The common example of these approaches is the Latent Semantic Analysis (LSA). However, the main limitation behind the corpus-based approaches, in contrary to the use of knowledge source, is that they become domain-dependent after being fed by specific corpus. This makes them too sensitive toward the domain of the answers.

Further research attempts showed different techniques for semantic analysis; in particular, the Neural-Network-based techniques. Such techniques aim at processing a set of token words as input to a neural network architecture for the purposed of outputting distinctive embedding for each term. Such an embedding would capture the semantic, lexical and other important features of the word and represent them in a vector. This vector then would be utilized for other tasks such as the AES. The common example of these techniques is the Word2Vec architecture [45]. The problem of this architecture is represented by the need to train the model on large text, meanwhile, fine tune the parameters of the network; otherwise, it would generate inaccurate embedding.

To solve the aforementioned problem, further researches have presented a pre-trained model in which the model is being trained on large text and its parameters are fine tuned. The literature showed the utilization of a pre-trained Word2Vec models [11] along with another pre-trained model known as Glove ([9]; [26]; [27]; [28]; [45]; [46]). The main limitation behind these architectures is that they only work with word-level which obstructs them from working on document/sentence-level. Since the AES task is mainly depending on sentence/document answers, the traditional word embedding architectures seem insufficient.

Other studies have utilized much more sophisticated architecture to handle document/sentence embedding such as Doc2Vec [20] or Google Sentence Encoder (GSE) [21]. However, these architectures suffer from a common limitation known as 'out-of-vocabulary'. This problem occurs when an embedding architecture is being tested with a word that has no embedding vector within its training model.

In fact, the embedding techniques are considered as the state-of-the-art techniques that have shown remarkable performance in the AES task. Therefore, it is a significant effort to overcome the 'out-of-vocabulary' problem in order to enhance the semantic analysis which indeed would reflect on improving the essay answer scoring.

A remarkable overcome for the aforementioned problems has been depicted by the emergence of Bidirectional Encoder Representation from Transformer (BERT) architecture [13]. Such an architecture has the ability to overcome the averaging embedding for larger text units (e.g., document and paragraph) by utilizing a pretrained embedding that works by treating the sentence as a combination of words with fixed an indexed embedding. In addition, it has the ability to overcome 'out-of-vocabulary' problem by dividing unseen words into an indexed and recognized words from its vocabulary repository. BERT has two models; language modeling and fine-tuning. The first model aims at understand the language of a text and its latent contextual information. Whereas, the second model aims at accommodating the desired task such as question answer, document classification or ranking.

However, multiple recent researches showed that BERT architecture has non-outstanding performance on AES task compared to techniques ([16]; [30]; [43]). Although BERT showed magnificent performance in problems like question answering, its architecture failed to give an accurate scoring for an answer. The reason behind such failure lies on a problem called 'catastrophic forgetting' where its language model forgets significant contextual information that impact the scoring. In addition, BERT suffers from the tremendous extent of parameters where around 60 million parameters represent a highly computational requirement [35].

According to [7], the application of most sophisticated embedding techniques including BERT on AES task is still lacking of latent rubrics. This is because the scoring task is still challenging for humans themselves. Hence, the language modeling architectures show an outstanding ability of capturing semantic of text. Yet, it is still lacking the writing style or structural features.

V. Conclusion

This paper has provided a review on the feature analysis used for either supervised or unsupervised AES. Within such a review, a taxonomy has been represented for the feature analysis which included four main types; Morphology, Frequencies, Structure, and Semantic. Inside each type, various subcomponents and approaches have been illustrated. After that, a critical review has been provided on the recent AES studies by linking each feature analysis type to these studies. The finding of such a critical analysis showed that the traditional morphological analysis of the essay answer would

lack the semantic analysis. Whereas, utilizing a semantic knowledge source such as ontology would be restricted to the domain of the essay answer. Similarly, utilizing semantic corpus-based techniques would be impacted by the domain of the essay answer as well. On the other hand, using essay structural features and frequencies alone would be insufficient, but rather as an auxiliary to another semantic analysis technique would bring promising results. The state-of-the-art in AES researches concentrated on neural-network-based-embedding techniques. Yet, the major limitations of these techniques are represented as (i) finding an adequate sentence-level embedding when using models such as Word2Vec and Glove, (ii) 'out-of-vocabulary' when using models such as Doc2Vec and GSE, and lastly, (iii) 'catastrophic forgetting' when using BERT model.

ACKNOWLEDGMENT

This study is supported by the University Kebangsaan Malaysia (UKM).

REFERENCES

[1] Ajetunmobi, S. A., & Daramola, O. (2017, 29-31 Oct. 2017). Ontology-based information extraction for subject-focussed automatic essay evaluation. Paper presented at the 2017 International Conference on Computing Networking and Informatics (ICCNI).

[2] Al-Jouie, M. F., & Azmi, A. M. (2017). Automated Evaluation of School Children Essays in Arabic. Procedia Computer Science, 117, 19-22. doi:https://doi.org/10.1016/j.procs.2017.10.089.

[3] Alqahtani, A., & Alsaif, A. (2019, 10-12 Dec. 2019). Automatic Evaluation for Arabic Essays: A Rule-Based System. Paper presented at the 2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT).

[4] Alshaikhdeeb, B., & Ahmad, K. (2015). Integrating correlation clustering and agglomerative hierarchical clustering for holistic schema matching. Journal of Computer Science, 11(3), 484-489.

[5] Azmi, A. M., Al-Jouie, M. F., & Hussain, M. (2019). AAEE–Automated evaluation of students' essays in Arabic language. Information Processing & Management, 56(5), 1736-1752.

[6] Azmi, A. M., Al-Jouie, M. F., & Hussain, M. (2019). AAEE – Automated evaluation of students' essays in Arabic language. Information processing & management, 56(5), 1736-1752. doi:https://doi.org/10.1016/j.ipm.2019.05.008.

[7] Beseiso, M., & Alzahrani, S. (2020). An Empirical Analysis of BERT Embedding for Automated Essay Scoring.

[8] Blanchard, D., Tetreault, J., Higgins, D., Cahill, A., & Chodorow, M. (2013). TOEFL11: A corpus of non ‐ native English. ETS Research Report Series, 2013(2), i-15.

[9] Chen, Z., & Zhou, Y. (2019, 25-28 May 2019). Research on Automatic Essay Scoring of Composition Based on CNN and OR. Paper presented at the 2019 2nd International Conference on Artificial Intelligence and Big Data (ICAIBD).

[10] Choi, E., He, H., Iyyer, M., Yatskar, M., Yih, W.-t., Choi, Y., . . . Zettlemoyer, L. (2018). Quac: Question answering in context. arXiv preprint arXiv:1808.07036.

[11] Cozma, M., Butnaru, A. M., & Ionescu, R. T. (2018). Automated essay scoring with string kernels and word embeddings. arXiv preprint arXiv:1804.07954.

[12] Crossley, S. A., & Kyle, K. (2018). Assessing writing with the tool for the automatic analysis of lexical sophistication (TAALES). Assessing Writing, 38, 46-50. doi:https://doi.org/10.1016/j.asw.2018.06.004.

[13] 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.

[14] Filho, A. H., Concatto, F., Nau, J., Prado, H. A. d., Imhof, D. O., & Ferneda, E. (2019). Imbalanced Learning Techniques for Improving the Performance of Statistical Models in Automated Essay Scoring. Procedia Computer Science, 159, 764-773. doi:https://doi.org/10.1016/j.procs.2019.09.235.

[15] Filho, A. H., do Prado, H. A., Ferneda, E., & Nau, J. (2018). An approach to evaluate adherence to the theme and the argumentative structure of essays. Procedia Computer Science, 126, 788-797. doi:https://doi.org/10.1016/j.procs.2018.08.013.

[16] Fukuda, H., Tsunakawa, T., Oshima, J., Oshima, R., Nishida, M., & Nishimura, M. (2020). BERT-based Automatic Text Scoring for Collaborative Learning. Paper presented at the 2020 IEEE 9th Global Conference on Consumer Electronics (GCCE).

[17] Garner, J., Crossley, S., & Kyle, K. (2019). N-gram measures and L2 writing proficiency. System, 80, 176-187. doi:https://doi.org/10.1016/j.system.2018.12.001.

[18] Gomaa, W. H., & Fahmy, A. A. (2014). Automatic scoring for answers to Arabic test questions. Computer Speech & Language, 28(4), 833-857.

[19] Hasanah, U., Astuti, T., Wahyudi, R., Rifai, Z., & Pambudi, R. A. (2018, 13-14 Nov. 2018). An Experimental Study of Text Preprocessing Techniques for Automatic Short Answer Grading in Indonesian. Paper presented at the 2018 3rd International Conference on Information Technology, Information System and Electrical Engineering (ICITISEE).

[20] Hassan, S., Fahmy, A. A., & El-Ramly, M. (2018). Automatic Short Answer Scoring based on Paragraph Embeddings. International Journal of Advanced Computer Science and Applications, 9(10), 397-402.

[21] Hendre, M., Mukherjee, P., Preet, R., & Godse, M. (2020). Efficacy of Deep Neural Embeddings based Semantic Similarity in Automatic Essay Evaluation. International Journal of Computing and Digital Systems, 9, 1-11.

[22] Kopparapu, S. K., & De, A. (2016, 21-24 Sept. 2016). Automatic ranking of essays using structural and semantic features. Paper presented at the 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI).

[23] Kumar, V., Ramakrishnan, G., & Li, Y.-F. (2018). A framework for automatic question generation from text using deep reinforcement learning. arXiv preprint arXiv:1808.04961.

[24] Kyle, K. (2020). The relationship between features of source text use and integrated writing quality. Assessing Writing, 45, 100467. doi:https://doi.org/10.1016/j.asw.2020.100467.

[25] Kyle, K., & Crossley, S. (2016). The relationship between lexical sophistication and independent and source-based writing. Journal of Second Language Writing, 34, 12-24. doi:https://doi.org/10.1016/j.jslw.2016.10.003.

[26] Li, X., Chen, M., & Nie, J.-Y. (2020). SEDNN: Shared and enhanced deep neural network model for cross-prompt automated essay scoring. Knowledge-Based Systems, 210, 106491. doi:https://doi.org/10.1016/j.knosys.2020.106491.

[27] Li, X., Chen, M., Nie, J., Liu, Z., Feng, Z., & Cai, Y. (2018). Coherence-Based Automated Essay Scoring Using Self-attention Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data (pp. 386-397): Springer.

[28] Liu, T., Ding, W., Wang, Z., Tang, J., Huang, G. Y., & Liu, Z. (2019). Automatic short answer grading via multiway attention networks. Paper presented at the International Conference on Artificial Intelligence in Education.

[29] Mathias, S., & Bhattacharyya, P. (2018). ASAP++: Enriching the ASAP automated essay grading dataset with essay attribute scores. Paper presented at the Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018).

[30] Mayfield, E., & Black, A. W. (2020). Should You Fine-Tune BERT for Automated Essay Scoring? Paper presented at the Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications.

[31] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. Paper presented at the Advances in neural information processing systems.

[32] Mohamed, O. J., ZAKAR, N. A., & Alshaikhdeeb, B. (2019). A combination method of syntactic and semantic approaches for

classifying examination questions into Bloom's taxonomy cognitive. Journal of Engineering Science and Technology, 14(2), 935-950.

[33] Montgomery, D. C., Peck, E. A., & Vining, G. G. (2015). Introduction to linear regression analysis: John Wiley & Sons.

[34] Nadeem, F., Nguyen, H., Liu, Y., & Ostendorf, M. (2019). Automated Essay Scoring with Discourse-Aware Neural Models. Paper presented at the Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications.

[35] Ormerod, C. M., Malhotra, A., & Jafari, A. (2021). Automated essay scoring using efficient transformer-based language models. arXiv preprint arXiv:2102.13136.

[36] Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P., . . . Iyengar, S. (2018). A survey on deep learning: Algorithms, techniques, and applications. ACM Computing Surveys (CSUR), 51(5), 1-36.

[37] Pramukantoro, E. S., & Fauzi, M. A. (2016, 15-16 Oct. 2016). Comparative analysis of string similarity and corpus-based similarity for automatic essay scoring system on e-learning gamification. Paper presented at the 2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS).

[38] Ratna, A. A. P., Khairunissa, H., Kaltsum, A., Ibrahim, I., & Purnamasari, P. D. (2019, 2-3 Oct. 2019). Automatic Essay Grading for Bahasa Indonesia with Support Vector Machine and Latent Semantic Analysis. Paper presented at the 2019 International Conference on Electrical Engineering and Computer Science (ICECOS).

[39] Ratna, A. A. P., Luhurkinanti, D. L., Ibrahim, I., Husna, D., & Purnamasari, P. D. (2018, 21-22 Sept. 2018). Automatic Essay Grading System for Japanese Language Examination Using Winnowing Algorithm. Paper presented at the 2018 International Seminar on Application for Technology of Information and Communication.

[40] Ratna, A. A. P., Noviaindriani, R. R., Santiar, L., Ibrahim, I., & Purnamasari, P. D. (2019, 22-24 July 2019). K-Means Clustering for Answer Categorization on Latent Semantic Analysis Automatic Japanese Short Essay Grading System. Paper presented at the 2019 16th International Conference on Quality in Research (QIR): International Symposium on Electrical and Computer Engineering.

[41] Ratna, A. A. P., Santiar, L., Ibrahim, I., Purnamasari, P. D., Luhurkinanti, D. L., & Larasati, A. (2019, 23-25 Oct. 2019). Latent Semantic Analysis and Winnowing Algorithm Based Automatic Japanese Short Essay Answer Grading System Comparative Performance. Paper presented at the 2019 IEEE 10th International Conference on Awareness Science and Technology (iCAST).

[42] Ridley, R., He, L., Dai, X., Huang, S., & Chen, J. (2021). Automated Cross-prompt Scoring of Essay Traits.

[43] Rodriguez, P. U., Jafari, A., & Ormerod, C. M. (2019). Language models and Automated Essay Scoring. arXiv preprint arXiv:1909.09482.

[44] Tashu, T. M. (2020, 3-5 Feb. 2020). Off-Topic Essay Detection Using C-BGRU Siamese. Paper presented at the 2020 IEEE 14th International Conference on Semantic Computing (ICSC).

[45] Wang, Z., Liu, J., & Dong, R. (2018, 23-25 Nov. 2018). Intelligent Auto-grading System. Paper presented at the 2018 5th IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS).

[46] Zhang, H., & Litman, D. (2019). Co-attention based neural network for source-dependent essay scoring. arXiv preprint arXiv:1908.01993.

[47] Zhu, X. (2006). Semi-supervised learning literature survey. Computer Science, University of Wisconsin-Madison, 2(3), 4.

[48] Zou, W. Y., Socher, R., Cer, D., & Manning, C. D. (2013). Bilingual word embeddings for phrase-based machine translation. Paper presented at the Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing.