An Ensemble-Based Model to Improve the Accuracy of Automatic Short Answer Grading

Mostafa Mohamed Saeed
Faculty of Computer Science
MSA University
Giza, Egypt
mostafa.mohamed52@msa.edu.eg

Wael H. Gomaa
Faculty of Computer and Artificial Intelligence, Beni-Suef University
Faculty of Computer Science, MSA University
Egypt
Wahassan@msa.edu.eg

2022 2nd International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC) | 978-1-6654-6677-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/MIUCC55081.2022.9781737

***Abstract*—Since 1966 much research has been done on the automatic grading of student answers task, and it was divided into short answer grading and essay scoring. In this paper, on the short answer grading challenge, we are working with text similarity approaches that are being classified into string, semantic (corpus and knowledge-based), and embedding text similarity approaches are the three types of text similarity techniques. On the Texas data-structure data set, different experiments were examined individually before being merged to give a maximum correlation result of 0.65.**

***Index Terms*—Automatic Scoring, Natural Language Processing, Machine Learning, Text Similarity**

I. Introduction

The intersection or mixture of computer science, linguistics, and machine learning is known as natural language processing (NLP). This area focuses on computer-human natural language communication, and NLP is all about training computers to interpret human language as effortlessly as people do. NLP has benefited greatly from recent advances in machine learning, particularly deep learning methods. The field is organized into several areas, the most prominent of which are: Natural Language Understanding (NLU) refers to a computer's capacity to understand what we say as clearly as humans, whereas Natural Language Generation (NLG) refers to the ability of a computer to generate natural language. Our major focus will be Natural Language Understanding, and these two components can perform a variety of tasks including automatic translation, named-entity recognition, summarization, connection extraction, sentiment analysis, audio recognition, and topic segmentation. NLP is a challenging field. There are a lot of factors that make this process more difficult, including the fact that there are hundreds of natural languages, each with its unique set of syntactic rules. Words can be ambiguous, altering their meaning depending on the context. Furthermore, ambiguous sentences may be a prevalent difficulty in NLP. Sentences and phrases that can be construed in two or more ways are said to as ambiguous. Reading a sentence without the context of the surrounding language is challenging for humans, so imagine how tough it is for a machine to understand this. POS (part of speech) tagging is one NLP approach that may be employed. Humans are easily able to rectify spelling errors. We have the capacity to distinguish between a misspelt word and its correctly spelled equivalent. A machine, on the other hand, will have a tougher problem comprehending a phrase that contains several spelling mistakes. To complete this demanding task, natural language processing methods were used. Another issue is identifying each individual sentence since phrases and sentences are made up of words that are combined. Phrase detection may be tough, and it's not as simple as looking for periods at the end of a sentence. Periods can be used in a variety of locations where they should represent the conclusion of a sentence, but they are often used as a shorthand for something else, such as Mrs. or Mr., or decimal numbers like 12.834. Furthermore, there are other issues that make working in the NLP field challenging and difficult. Text similarity measures play an effective and important role in all NLP related research and applications. Finding word similarity is a crucial aspect of text similarity, which is then utilized as a starting point for comparing sentences, paragraphs, and documents. Words can be compared to each other from two ways semantical and lexical. String based similarity is a metric that measures the distance between two or more text strings. From the most popular algorithms in string-based similarity Longest Common Substring (LCS) that calculate the similarity according to the length of characters that exist in both strings, Dameraue Levenshtein that is based on counting the minimum operations needed to transform one string to another [1], [2], Cosine similarity that is based on calculating the cosine angle between two vectors to determine their similarity. and Jaccard Similarity that is calculated according to the number of shared terms over the number of terms in the two comparable words [3]. For semantic similarity, approaches there are two types, Corpus-Based Similarity and Knowledge-Based Similarity. Corpus-Based Similarity is based on the similarity between two words that exist in a corpus. There are much algorithms for corpus-based similarity like "Hyper Analogue to language (HAL)" [4], [5] and "Latent Semantic Analysis (LSA)" [6]. Knowledge-Based similarity is based on calculating the semantic degree between two words according to a large semantic network like WordNet [7] database that groups each word in a unique sysnet that is an interlinked with all the information needed about each word and also its conceptual semantic

978-1-6654-6677-6/22/$31.00 ©2022 IEEE

and lexical relations. There are different semantic similarity techniques using WordNet. There are three algorithms that are based on information content which are "Resnik algorithm (res)" [8], "Lin algorithm (lin)" [9], and "Jiang Conrath algorithm (jcn)" [10]. The other three metrics are based on path length, "Leacock Chodorow algorithm (lch)" [11], "Wu Palmer algorithm (wup)" [12], and "Path Length" algorithm. Over the last few years, the educational system has undergone significant adjustments, particularly in the aftermath of the COVID-19 pandemic. As a result, the educational phases have shifted from being only between the student and the academic staff to be between them both but with technology that helps them in their overall education processes. As a result, the idea of implementing an automatic grading system has gained traction, owing to the ever-expanding educational community. Moreover, this approach will undoubtedly provide numerous benefits, such as assisting academics in decreasing their burden and allowing them to focus on other tasks. In addition to that, it will make the process of sending the real grades to the students more promptly rather than waiting for manual examiners to grade their answers one by one, and it will obtain unbiased grading results, ensuring that grading is always done in a standard formal manner. This study proposes an automatic scoring system for short answers that calculate an efficient and quick grade for the student answer based on different text similarity techniques. Automatic grading comprises reviewing and grading any student answer and awarding a specific grade depending on the contents. There are two sorts of automatic grading: essay scoring and short response grading, with each having its own system or methodology. ES is a challenging process to perform successfully and efficiently since it involves grading long essay questions based on content, grammar checking, correct run-ons, punctuation, and more other concepts of linguistics. The majority of ES approaches are used in the English language; however, ES may be characterized by other terms such as automated essay evaluation, automated essay evaluations, and automated writing scoring. Its execution necessitates the training of a model on hundreds of essay replies assessed by experts, after which the computer extracts certain particular aspects from these essays that will aid him in developing a model capable of predicting the human manual grade. Several software systems have been developed, including Project Essay Grader, which uses proxy metrics to assess essay quality, Intelligent Essay Assessor, which compares the semantic similarity of textual paragraphs information, and E- rater, which was developed by the Educational Testing Service and relies primarily on linear regression. The next sections will cover some related works for the same topic, data set description and example, our approach (which comprises five stages), implemented experiments and their outcomes, analysis of our results, and finally conclusion and future work.

II. Related work

Various approaches for implementing an automatic short response grading system have been presented in the previous ten years.

(Tulu et al.) [13] This research has proposed a new technique using a deep learning approach, which is MaLSTM and sense vectors obtained by SemSpace. The student answer is represented by a sysnet-based embedding of WordNet, and the reference answer is inputted into a parallel LSTM model, which leads to the transformation of these two representations into vectors and the use of Manhattan similarity to measure the similarity between the two vectors (student answer and reference answer). This technique was utilized and evaluated on the Texas data structure dataset, and they were able to reach a correlation of 0.95 by separating the data into train and test for each question. When they operate on the dataset as is, however, they get a correlation of 0.15. Furthermore, they said that LSTM does not take into account long-range dependencies, in contrast to the transformers that perform admirably in the automatic short response grading assignment.

(Süzen et al.) [14] This study focuses on automatic scoring of the students' answers since it is common in the United Kingdom, and they would want to provide feedback on the students' answers whether they are incomplete or incorrect. This study suggested using clustering, regression analysis, and hamming distance to find the similarity between the student answer and the given reference answer to solve the automatic short answer grading problem, but they didn't go into detail about the technicalities involved in getting their results. The authors evaluated their strategy on the Texas data structure data set, achieving a correlation of 0.81, and the authors said that semantics or synonyms were not taken into consideration in this approach.

(Gomaa and Fahmy) [15] In this study, the Ans2vec approach was suggested as a simple and efficient model for scoring brief answers. To convert the model and student responses into vectors. They make use of Ans2vec (Skip-thought vector technique). Skip-thought vectors are a sentence-level version of word2vec, in which the surrounding words is being predicted by the word2vec and the surrounding sentences is being predicted by the skip thought vectors. According to the authors, they evaluated their model on the Texas Data Structure data set and obtained a correlation of 0.63. They stated that working with the Texas dataset is tough because most researchers struggle to produce correlation results greater than 0.7, as can be demonstrated in this study.

(Pribadi et al.) [16] The first stage of this research will propose (Maximum Marginal Relevance (MMA), which will generate a reference answer from the given student answer, and the second stage will propose "GAN-Longest Common Subsequence (GAN-LCS)", which is an extension of the LCS that works on computing the similarity between two sentences of different lengths, with the output coefficient being the student grade. This approach was tested on the Texas Dataset, which yielded a 0.468 correlation score.

(Hassan et al.) [17] This study proposes using a paragraph embedding approach for both the student and reference answer, then computing the cosine similarity between them and using the result as a feature in a regression model to predict

the student grade. The vector representation of the student response and the reference answer might be generated in two ways. The first method was to create the vector by summing the word vectors in our text. To construct our vectors, we used the second way of training a deep learning model. All word embedding was done with Glove, Word2Vec, and fasttext, while all paragraph embedding was done with Skip- thoughts, Doc2Vec, and InfraSent. The author utilized the Texas dataset to test his method, and they were successful.

(S. Roy et al.) [18] The focus of this study was on vector-based approaches. They began by deleting all stop words from their data before computing the seven to seven similarity measures: block distance, JingConarth, DISCO, Lesk, Glove, Sense Vectors, and Word2Vec. They also employed sentence-to-sentence similarity metrics such as the text-to-text model, the Min-Max additive model, and the Vector Summa- tions Model. Finally, all of these methods were evaluated on the Texas dataset, with the greatest correlation value of 0.586.

(Magooda et al.) [19] His system is offered as a three-module sequence in this study. He began by pre-processing his data by eliminating stop words and doing lemmatization or stemming. Second, he will begin calculating certain similarity measures using the word-to-word approach, which will include string similarity techniques such as block distance. In addition to knowledge-based techniques like JiangConrath and Lesk algorithm, corpus-based techniques like DISCO, and the last type of similarity, which is the Vector representation, such as Word2Vec toolkit, Glove pre-trained model, and sense aware vectors, there is also the Vector representation, which includes Word2Vec toolkit, Glove pre-trained model, and sense aware vectors. Finally, the scaling module assigns a grade to the student between 0 and 5 on a scale of 1 to 5. This method was tried on a dataset from Texas, and he was able to reach a correlation of 0.550.

III. Data set

TABLE I: Example for a question and its model answer

Sample Question
In one sentence, what is the main idea implemented program by insertion sort?
Model Answer
Taking one array element at a time, from left to right, it inserts it in the right position among the already sorted elements on its left.

The ASAG task uses six datasets, starting with the CREG and CREE datasets (Meurers et al. 2011), the ASAP dataset that is from a Kaggle competition (Hewlett 2012), the SciEnts-Bank and Beetle datasets (Dzikovska et al., cited in Galhardi Brancher, 2018), and the Texas dataset from (Meurers et al.) The Texas dataset was chosen as the major source for implementing the ASAG out of those six datasets. The reason for choosing the Texas dataset is because it is the most difficult dataset to use as the main source. The Texas Dataset (Mohler 2011) contains 87 questions chosen from 10 assignments (each assignment has from four to seven questions) and two tests

TABLE II: Two student answers with manual evaluators grade

Student 1 Answer	**Grade/5**
Take a number and choose a pivot point and insert the number in the correct position from the pivot point..	**3/5**
Student 2 Answer	**Grade/5**
Insertion sort removes an element from the data, and inserts it at the correct position in the already sorted list.	**5/5**

(each with ten questions), all of which are connected to an introduction of computer science curriculum for undergrad students (Mohler et al. 2011). There are 2,442 replies from students. The data includes the average grades of the two manual graders that grades each answer on scale of 0 to 5. When it comes to grading, grader one is more tolerant than grader two. Grader one assigned a 4.43 average, whereas grader two assigned a 3.94 average. Because it was an augmentation of the dataset generated by the authors in 2009, this dataset is also known as Texas Extended Dataset (Mohler and Mihalcea 2009).

IV. Methodology

Our proposed model will pass through five stages as shown in fig (1). The first stage is the data set preparation. The second stage is the pre-processing stage. The third stage is the processing stage where multiple text similarity techniques will be implemented to compare the student answer with the same question given reference answer. The fourth stage is to grade the student answer. Then finally, the last stage is evaluating the proposed model grade according to the manual given grade by the examiner (educational staff).

A. Data set preparation

In the first stage as shown in fig (2), we have to extract everything separately and efficiently from our data. Starting with extracting the question itself, the student answer for each question, all the given reference answers for each question and finally the average grade of the two evaluators that will be our main target to reach.

B. Pre-Processing

In this phase as shown in as shown in fig (3), Most of the linguistic techniques are being applied. The first technique is converting all the letters of all given sentences (including the question itself or the question reference answer or the student answer) into lower case letters as in NLP the capital 'A' is not equal to a small 'a'. The second technique is performing the stemming which is converting each word is given into its root form or lemmatization which is the process of combining a different word's inflected forms so that they may be studied as a single item. Moreover, these two processes have to be done after splitting each sentence according to the spaces. The third technique is removing all stop words and special characters

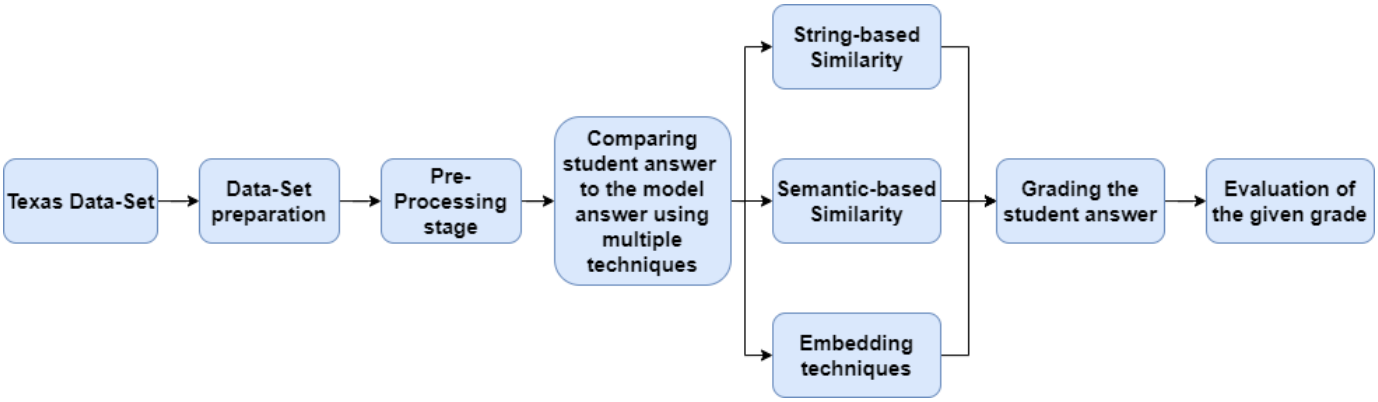

Fig. 1: General System Architecture

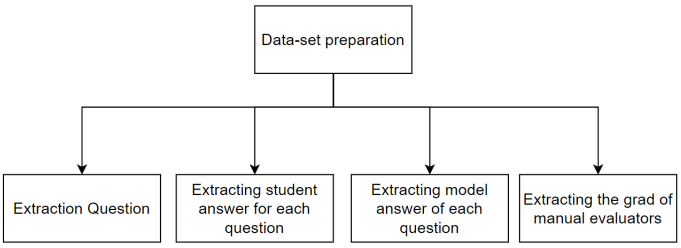

Fig. 2: Data set preparation processes

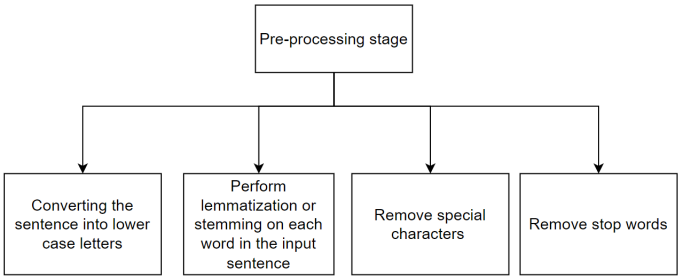

Fig. 3: Pre-Processing processes

in our sentences. And the final technique is converting all numeric numbers into alphabetical numbers.

C. Processing

In this phase, Multiple text similarity techniques and algorithms are going to be implemented and then combined together to form our Hybrid proposed model. For the string-based similarity algorithms, we will apply 168 string-based algorithm from Abydos library. For the Corpus-based similarity algorithms, Latent Semantic Analysis (LSA) algorithm will be implemented and for Knowledge-based similarity algorithms, five algorithms from WordNet similarity algorithms will be applied which are LI, LIN, WPATH and JCN. Finally, for Embedding techniques, we will be using Bert models, Glove models, Roberta, and more other models that convert our sentences into vectors and then apply some text similarity algorithms using their pre-trained models.

D. Predicting the grade

In this phase, our main target is to make our regression model got the ability to predict the quick and efficient grade for the student after merging all the similarity algorithms output that we have calculated while comparing the student answer to the reference answer.

E. Evaluation

In the final phase, the correlation metrics are used in this regression task to evaluate all the output grades predicted by our proposed model.

V. Experiment and Results

The environment that was used to implement this approach was on a laptop with core-i7 8th Gen, 16GB ram and GPU Nvidia GTX 1060 6GB.

TABLE III: Correlation results of some techniques without combination

Text-similarity technique	Correlation Result
Kuhns VI	0.488928
GiniII	0.487648
Iterative SubString coorelation	0.486732
Forbes II	0.481941
Fuzzy Wuzzy Token Set similarity	0.477305
Kuhns IV	0.473229
Tulloss S	0.466162
Cosine	0.420157

1) Experiments: Text-similarity strategies (algorithms) were applied independently without being integrated with any other algorithm in the early studies, but the results were not the best as shown in Table.(III) and Fig(4).

Our technique delivers superior and comparable outcomes in experimental evaluation on Texas data-structure data set as shown. The implemented tests depicted in Table (IV) are the most significant of all since they introduce our fundamental concept and goal of creating an ensemble model. For the first experiment, the student answer and the reference answer were compared using string similarity approaches, with lower case

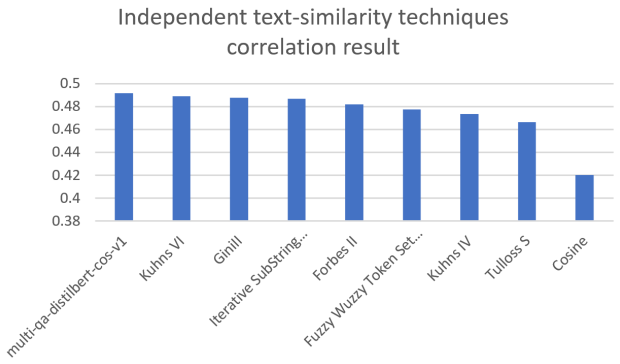

Fig. 4: Independent text similarity techniques correlation result

TABLE IV: Experiments

Pre-processing						Text-similarity				Corr.
Lower Case	Remove stop words	Remove special characters	Number to strings	Lemma	Stemming	String similarity	Semilar	Wordnet	Embedding models	Correlation results
√	-	-	-	√	-	√	√	√	√	0.6514
√	-	-	-	√	-	√	√	√	-	0.6105
√	-	-	-	√	-	√	-	-	-	0.5946
√	-	-	-	-	-	√	-	-	-	0.5923
√	-	-	-	-	-	√	-	-	-	0.5825
√	-	-	-	-	√	√	-	-	-	0.577

letters and stemming procedures applied to each word in both phrases, yielding a 0.577 correlation. The student answer and the reference answer were compared using string similarity methodologies, but without any processing techniques on the student answer or the manual reference answer, yielding a 0.5825 correlation. For the third experiment, the student answer and the reference answer were compared using string similarity methodologies, with lower case letters and stemming techniques used for each word in both phrases, yielding a 0.5923 result. And this demonstrates that the pre-processing stage is one of the most important aspects that must be applied well in this technique. The concept of building an ensemble model was used in the fourth and fifth experiments, where lower case letters technique and lemmatization were used for all sentences, but the fourth experiment was a combination of string and semantic similarity techniques with a correlation of 0.6105, while the fifth experiment with the addition of only some embedding text similarity techniques like ”bert-base-nli-mean- tokens” model. This is a sentence transformers Bert model that has been pre-trained on one million sentence pairings with a batch size of 16 and a learning rate of 2e-5. It has a maximum sequence length of 128 characters and a word

embedding dimension of 728. which results in a correlation equal to 0.6514.

For the second experiment, The student answer and the reference answer were compared using string similarity approaches, but without any pre-processing technique on the student answer or the manual model answer, which result for 0.5825 correlation. In this approach K-fold was implemented as our validation technique, the data is separated into different k-subsets of data, with each k subset working as a test set and the k-1 subset acting as a training set, and a holdout process being performed k-times. (k=10 in our implementation). Texas data set. Moreover, different classifiers have been implemented to our data and the Random Forest classifier was the best one each time for the correlation final result.

2) Results: Our proposed hybrid approach has achieved better or more competitive results in the experimental evaluation on the Texas benchmarking data set as shown in Table (V) and fig (5). Our model achieved 0.65 correlation result compared to all the other systems.

TABLE V: Results of Texas data set

System	Correlation
Chaturvedi and Basak [20]	0.805
Our proposed Model	**0.65**
Gomaa, W. H., & Fahmy [15]	0.63
S. Roy et al [18]	0.57
Hassan et al [17]	0.569
Magooda et al [19]	0.55
Mohler et al [21]	0.52
Gomaa et al [22]	0.50
Basak et al [23]	0.365
Tulu et al [13]	0.15

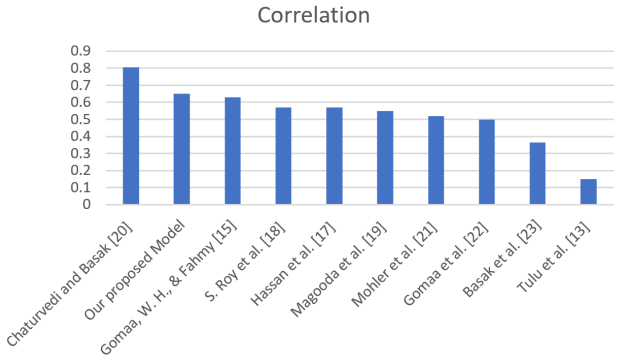

Fig. 5: Comparable Correlation Results

VI. Result Analysis

As shown in Table (III) the best text-similarity technique correlation was 0.488928 utilizing the Kuhns IV algorithm which indicates that employing text similarity, semantics, or even embedding techniques individually without combining them will not yield the best correlation results.

Moreover, as shown in Table (V), the combination between the 168 string similarity algorithm has increased the string similarity techniques' overall correlation results. In addition to that, adding standard semantic similarity techniques and embedding text similarity techniques improved our model outcomes, and ultimately, embedding text similarity techniques

models increased our hybrid model's confidence and accuracy compared to other simple and complex approaches. The ensemble model delivers more competitive results since each individual algorithm is powerful in its own way, and by combining them, the model becomes much more efficient.

VII. Conclusion and future work

Automatic Short Answer Grading has been one of the most important tasks that need to be implemented in an accurate, efficient and quick way in real life. In comparison to other complicated implementations, this study has shown that using simple procedures which are string similarity or semantic or embedding similarity techniques or the combination between them in one hybrid approach without the usage of bert models can result for a very efficient correlation result.

Future work will be concerned with different approaches. Different data sets will be applied on our model like the Beetle and SciEntsBank data set or Corpus of Reading comprehension exercises in German data set (CREG) or the Corpus of Reading Comprehension Exercises (CREE) data set. From other approach which is testing our proposed model on different language data sets like Arabic data sets. Also, during analyzing Texas data structure data set, it was noticed that there are many numbers, mathematical equations and programming code which need handcrafting techniques or deep learning approaches either using transformers or deep neural networks or even both merged together that will lead for more efficient results in this field specifically.

References

[1] Patrick AV Hall and Geoff R Dowling. Approximate string matching. *ACM computing surveys (CSUR)*, 12(4):381–402, 1980.

[2] James L Peterson. Computer programs for detecting and correcting spelling errors. *Communications of the ACM*, 23(12):676–687, 1980.

[3] Paul Jaccard. Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bull Soc Vaudoise Sci Nat*, 37:547–579, 1901.

[4] Kevin Lund. Semantic and associative priming in high-dimensional semantic space. In *Proc. of the 17th Annual conferences of the Cognitive Science Society, 1995*, 1995.

[5] Kevin Lund and Curt Burgess. Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior research methods, instruments, & computers*, 28(2):203–208, 1996.

[6] Thomas K Landauer and Susan T Dumais. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, 104(2):211, 1997.

[7] George A Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J Miller. Introduction to wordnet: An on-line lexical database. *International journal of lexicography*, 3(4):235–244, 1990.

[8] Philip Resnik. Using information content to evaluate semantic similarity in a taxonomy. *arXiv preprint cmp-lg/9511007*, 1995.

[9] Dekang Lin. Extracting collocations from text corpora. In *First workshop on computational terminology*, pages 57–63. Citeseer, 1998.

[10] Jay J Jiang and David W Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. *arXiv preprint cmp-lg/9709008*, 1997.

[11] Claudia Leacock and Martin Chodorow. Combining local context and wordnet similarity for word sense identification. *WordNet: An electronic lexical database*, 49(2):265–283, 1998.

[12] Zhibiao Wu and Martha Palmer. Verb semantics and lexical selection. *arXiv preprint cmp-lg/9406033*, 1994.

[13] Cagatay Neftali Tulu, Ozge Ozkaya, and Umut Orhan. Automatic short answer grading with semspace sense vectors and malstm. *IEEE Access*, 9:19270–19280, 2021.

[14] Neslihan Süzen, Alexander N Gorban, Jeremy Levesley, and Evgeny M Mirkes. Automatic short answer grading and feedback using text mining methods. *Procedia Computer Science*, 169:726–743, 2020.

[15] Wael Hassan Gomaa and Aly Aly Fahmy. Ans2vec: A scoring system for short answers. In *International Conference on Advanced Machine Learning Technologies and Applications*, pages 586–595. Springer, 2019.

[16] Feddy Setio Pribadi, Adhistya Erna Permanasari, and Teguh Bharata Adji. Short answer scoring system using automatic reference answer generation and geometric average normalized-longest common subsequence (gan-lcs). *Education and Information Technologies*, 23(6):2855–2866, 2018.

[17] Sarah Hassan, Aly A Fahmy, and Mohammad El-Ramly. Automatic short answer scoring based on paragraph embeddings. *International Journal of Advanced Computer Science and Applications*, 9(10):397–402, 2018.

[18] Shourya Roy, Sandipan Dandapat, Ajay Nagesh, and Yadati Narahari. Wisdom of students: A consistent automatic short answer grading technique. In *Proceedings of the 13th International Conference on Natural Language Processing*, pages 178–187, 2016.

[19] Ahmed Ezzat Magooda, Mohamed Zahran, Mohsen Rashwan, Hazem Raafat, and Magda Fayek. Vector based techniques for short answer grading. In *The twenty-ninth international flairs conference*, 2016.

[20] Bhuvnesh Chaturvedi and Rohini Basak. Automatic short answer grading using corpus-based semantic similarity measurements. In *Progress in Advanced Computing and Intelligent Engineering*, pages 266–281. Springer, 2021.

[21] Michael Mohler, Razvan Bunescu, and Rada Mihalcea. Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 752–762, 2011.

[22] Wael H Gomaa and Aly A Fahmy. Short answer grading using string similarity and corpus-based similarity. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 3(11), 2012.

[23] Rohini Basak, Sudip Kumar Naskar, and Alexander Gelbukh. Short-answer grading using textual entailment. *Journal of Intelligent & Fuzzy Systems*, 36(5):4909–4919, 2019.