Grade It: A Quantitative Essay Grading System

DR. SINDHU
Department of Computing
SRM Institute Science and Technology
Kattankulathur, Tamil Nadu

ROOPCHAND REDDY VANGA
Department of Computer Science and Engineering
SRM Institute Science and Technology
Kattankulathur, Tamil Nadu
rv8364@srmist.edu.in

M.S. BHARATH
Department of Computer science and Engineering
SRM Institute Science and Technology
Kattankulathur, Tamil Nadu
Bs6396@srmist.edu.in

Abstract*—Automated writing evaluation system that employs automated essay scoring technologies Generates a rating to the writings which help students with self-assessment. It provides an efficient and easy way for grading Essays, which usually takes abundance of time to be evaluated by human graders, this can be greatly useful for educational institutions like schools and colleges. It provides ratings for essays based on the grammatical errors and topic relevancy by using specific tools implemented in the auto grading system. The generated result will help students grading essays which helps students in self-assessment, this will be useful for the student to understand the mistakes that he makes which will be pinpointed by the system in matter of time, not only that the system also displays the strengths of the user, system will improve the accuracy compared to already existing automated grading system. The system uses scores from multiple aspects (handcraft features, coherence score, Prompt-relevant score and semantic score). these key features illustrated above will provide enough information on the essay for the student

1.INTRODUCTION

As the amount of data continues to grow at an exponential rate, it is increasingly imperative to find ways to extract valuable insights from it in a timely and efficient manner. Furthermore, in educational settings, manually grading assignments for large classes can be a daunting task that consumes a significant amount of time for both educators and students. The shift to online learning brought about by the COVID-19 pandemic has further highlighted the need for an efficient grading system, particularly for essays where various elements such as grammar, spelling, and semantics must be closely evaluated. Implementing ML-based grading systems can not only save time but also provide more accurate and objective evaluations, allowing educators to focus on other important tasks and allowing students to receive their grades in a timely manner.

Despite the current limitations of AI-based writing evaluation (AWE) systems, they are still widely used by researchers and academics due to their potential to save time and provide more accurate and objective evaluations. However, the average accuracy of open-source AWEs used in the field is around 75%, and many of these systems are limited in their capabilities. AWEs are commonly used in educational settings to grade competitive exams like the TOEFL, GRE, and other exams from ETS. However, the low reliability of these systems and their lack of comprehensive grading can make them less useful for educators and students. The model aims to address these limitations by increasing the accuracy of an existing AWE model and making it more reliable for frequent use by educational institutions. We aim to detect writing errors by comprehending on various elements such as coherence, argumentation, and overall quality of the essay. By increasing the accuracy and accessibility of AWEs, we hope to make them a more valuable tool for educators and students, allowing them to save time and receive more accurate and detailed grading on their writing.

The project's objective is to enhance the grading process for students by developing a system that can provide more accurate and efficient results. This system utilizes natural language processing techniques and multiple tools such for structure checking, spelling and grammar checking, punctuation mark checking, and stop-word removal to analyse and grade essays. The system is trained by using pre-existing data and grading the models based on it. Additionally, the system also checks for relevance between the written essay and the assigned topic before grading. The accuracy of the model will be evaluated by comparing its results with real-world manual evaluations done by professors and instructors.

Our project's report begins with an abstract that provides a general overview of the topic, followed by an introduction that gives a more in-depth explanation of the topic and the reasons for choosing it. We then

REFERENCE ID	FEATURES	ALGORITHM USED	UNADDRESSED FEATURES
[12]	Use of handcrafted features like Checking punctuation, Missing letters, spelling mistakes etc.	Stemming and String based Algorithm	In terms of Coherence and semantics it's not the best model.
[15]	Uses text vectorization and Semantic analysis	Bi-LSTM-CRF model.	Doesn't detect adverbial essay and essays with permuted sentences.
[8]	Scores different aspects of an assay and merges them	CNN+LSTM	Data used in this model is scarce, hence hard to find.
[18]	Focusses on critical words and analyse the logic semantic relationship	LSTM Model	Variety of dataset available, hence hard to go with particular
[10]	uses the BERT for embedding, along with handcrafted features to predict the score.	Supervised Learning Algorithm	BERT model is slow to train due to weight to update, also its an expensive model

provide an overview of the steps and components used in creating the system. We discuss the current existing systems and their shortcomings, which we aim to address in the following sections. The methodology and experimentation, along with their results, are discussed in depth in the fourth section of the report.

2. RELATED WORK

The model is that is being developed is with the information gathered from other systems unaddressed features, advantages and their outcomes. An essay system developed in Saudi Arabian researchers [12] provided a wide variety of handcrafted features such as punctuation, checking for missing letters, spell check etc, they were successfully able to grade the essay with decent results but the model lacks coherence and semantics grading of the essay. Hence, handcrafted features were referred from their research work.

Another system had a good coherence grading, but the data set provided by the system was very less and the to run a essay system like the one that is provided by the institutions the system need a dataset that is huge. Most of the systems or projects used the vectorization as the main feature in the grading system, as it processes human language and understands meaning and context, vectorizations has been in use since the first computers were built it has been successful in various domains.

A hybrid model [8] uses BERT* model, it is mainly used in semantic analysis. which has the ability to process large amount of text and language. Which is very helpful while learning of pre -trained models, but this model also has its disadvantages as the model require large amount of data set to train and weight to update. These features will provide a better grading system for the user but then they also have their disadvantages, the purpose of this review is to identify the drawbacks of other working models and try to update the current model with the features and try to provide a more efficient way of grading.

3. GRADE IT: METHODOLOGY

In this section, the proposed methodology will be introduced to argue how both handcrafted features and deep-encoding features with semantics are used for training in our essay grader. Briefly, as shown in the figure 3.1 during the first stage, post minimal cleaning of the data, hand crafted features scores semantic score and prompt relevance score are calculated and in the second stage all scores are concatenated to give a final score. The final score, though just a number, is an amalgam of a broad set of features that are considered in manual grading process. Additionally, tools used in finding few handcrafted features such as punctuation marks, spelling errors are used to print respective types of errors.

A) Data cleaning

The data set used in the project is Automated student assessment price (ASAP). Essays in this dataset has tagged labels such as "@acb" which must be removed

to make the data conducive for finding handcrafted features. The scores or the 'y' label are different for each set of the essays causing inconsistencies hence all scores must be normalized and can be projected back to actual scores before printing the output.

No	Features
1	Character's average and range of word lengths.
2	Characters average and range of sentence lengths in words
3	Essay length in words and characters
4	Punctuation marks
5	word count for original essays
6	average amount of clauses per sentence
7	Mean clause length
8	Maximum number of clauses of a sentence in an essay

Table 3.1 features explained

B) Handcrafted features

In the table 3.1 are few handcrafted features used in [8]. Some of the 8 essay types in ASAP data such essay set-2 are not only scored based on the writing applications but also the language conventions which encapsulates features such as spelling mistakes, punctuation errors, grammatical errors, paragraphing, word count etc. Generally, low scored essays either contain very few words and sentences or be very lengthy. This implies writer's subpar skills in writing concisely, features like vocab size i.e., the number of unique words in the essay helps us understand the writer's vocabulary skill. For grammar, punctuation and spell check it only makes sense to use some existing grammar error correction systems or tools as they check for spelling errors, uncased letters, context, punctuation word repetition etc. In the model languagetool python, as its one of the most accurate systems [3], and ginger it to collect handcrafted features.

C) Sentence embedding/vectorization

It's not exaggeration to say that Google's Bidirectional Encoder Representations and transform (BERT) gave state of the art embedding results and proved itself to be better than word2vec in NLP tasks such as text classification [12]. Given the achievements of BERT, sentence embedding is performed on the stop words removed texts using $BERT_{base}$[2]. $BERT_{base}$ is a pretrained BERT model trained that has 768 dimensions and 12 encoder layers. "Pooled output" of the BERT result will be used to represent text into a multi-dimensional embedding.

pooled output: Is representation/embedding of CLS token passed through some more layers Bert pooler, linear/dense and activation function. It is recommended to use this pooled output as it contains contextualize information of whole sequence.

D) Semantic score

Deep semantic features are essential to analyse the semantic soundness of the essay and also for prompt relevance in prompt dependant tasks. Sequential model is used with LSTM to map our data into a low-dimensional embedding and pass it to the dense layer for scoring the essays.

For every essay e_x = {s1, s2, · · · , sm}, where s_k indicates the k th ($1 \leq k \leq m$, $s_k \in R^d$) embedded sentence in the essay and d = 768 means the length of sentence embedding. The encoding process of LSTM is described as follows [8]:

$$i = \sigma[W_i \,.\, s_t + U_I.\, h_{t-1} + b_i]\; c \quad - \; 1$$

$$f_t = \sigma\left[W_f \,.\, s_t + U_f.\, h_{t-1} + b_f\right] \quad - \; 2$$

$$C_t^{\sim} = \sigma[W_c.\, s_t + U_c.\, h_{t-1} + b_c] \quad - 3$$

$$C_t = it \,^{\circ} c_t^{\sim} + f_t \,^{\circ} c_{t-1} \quad - 4$$

$$0_t = \sigma[W_0 \,.\; S_t + U_0 \,.\; h_{t-1} + b_0] - 5$$

$$h_t = 0_t \,^{\circ} tanh(c_t) \quad - 6$$

h_t - hidden state of sentence st. (W_x, U_x) where x=(I,f,c,o) are the weight matrices for the input, forget , candidate and output gates respectively. b stands for the bias vectors of the specific gates. sigmoid function= σ and ◦ means element-wise multiplication. Hence, for every essay, we will get the hidden state set H = {h_1, h_2, · · ·, h_m}. H_m is passed to dense layer to convert to scalar value. The values from dense layer output are then projected back to their respective ranges according to ASAP dataset as has different sets of essays with different scoring range mentioned in Table 3.2.

Prompt	Essay	Avg Length	Score Range
1	1783	350	2-12
2	1800	350	1-6
3	1726	150	0-3
4	1772	150	0-3
5	1805	150	0-4
6	1800	150	0-4
7	1569	150	0-30
8	723	650	0-60

Table 3.2 grading values

E) Prompt relevancy score

This score shows how relevant the essay is to the question/prompt. Just like semantic score Sequential and LSTM model is used. Prompt = {s_1 ……..,s_k} and essay={s_1 ……..,s_k} where s refers to sentence are combined into one set of sentences. Now the exact same procedure followed for semantic score will be followed here i.e., after LSTM hidden layer data is fed

to dense layer which gives a scalar output scaled to 0-1 range and score 0 is received by essays that are irrelevant. The scores are projected back to their actual score ranges according to the dataset.

F) Training and evaluation metrics

It is known that essay sets ASAP dataset have different range of scoring so to be consistent all scores in each set in the training dataset are scaled down to 0-1 scale and will be scaled back to original values before testing phase. The LSTM neural network used in our model is mono layered with 1024 as the size of hidden layer. It is known that the performance of multi layered and bidirectional LSTM model is subpar from [8]. A dropout proportion of 0.5 is set in order to avoid overfitting of the model. The epochs are set to 50 and are compiled 5 times once for each fold of cross validation.

5.RESULTS DISCUSSION

For evaluation we use kappa score as the metric. In quantitative grading agreement between scores is more important than similarity between scores. No two graders are expected to provide similar scores but what's expected is agreement i.e., for example in case of 3 grader's scores such as 7.5/10, 8/10 and 7.9/10 are expected as they all are considered as positive scores but scores such as 8/10, 4/10, 9/10 are not in agreement so it may lead to ambiguity. To put it in perspective the AWE is expected to provide scores that are in agreement with actual human grader's scores to prove accuracy.

```
130/130 [==============================] - 3s 23ms/step - loss: 0.9579 - accuracy: 0.1686 - mae: 0.7423
Epoch 41/50
130/130 [==============================] - 3s 23ms/step - loss: 0.9710 - accuracy: 0.1689 - mae: 0.7464
Epoch 42/50
130/130 [==============================] - 3s 22ms/step - loss: 0.9195 - accuracy: 0.1693 - mae: 0.7261
Epoch 43/50
130/130 [==============================] - 3s 22ms/step - loss: 0.9454 - accuracy: 0.1690 - mae: 0.7376
Epoch 44/50
130/130 [==============================] - 3s 22ms/step - loss: 0.9441 - accuracy: 0.1692 - mae: 0.7345
Epoch 45/50
130/130 [==============================] - 3s 22ms/step - loss: 0.9109 - accuracy: 0.1690 - mae: 0.7199
Epoch 46/50
130/130 [==============================] - 3s 22ms/step - loss: 0.9392 - accuracy: 0.1683 - mae: 0.7340
Epoch 47/50
130/130 [==============================] - 3s 23ms/step - loss: 0.9084 - accuracy: 0.1687 - mae: 0.7180
Epoch 48/50
130/130 [==============================] - 3s 22ms/step - loss: 0.9313 - accuracy: 0.1686 - mae: 0.7309
Epoch 49/50
130/130 [==============================] - 3s 24ms/step - loss: 0.8928 - accuracy: 0.1692 - mae: 0.7150
Epoch 50/50
130/130 [==============================] - 3s 23ms/step - loss: 0.9068 - accuracy: 0.1686 - mae: 0.7156

Kappa Score: 0.9282690023079073
```

Fig. 5.1- final kappa score after 5 folds

For comparisons we used results sourced from TSL (Two stage learning) [8] which used CNN, CNN+LSTM and LSTM algorithms and Bayesian Linear Ridge Regression (BLRR) and Support Vector Regression (SVR) which uses domain adaptation method [14] and various other features such as parts of speech, and general handcrafted features. These models are chosen for comparison as they used a vast set of features for grading, proved to be reliable with high kappa scores. Table 5.1 and Fig 5.2 shows us how our model compares with other reliable models.

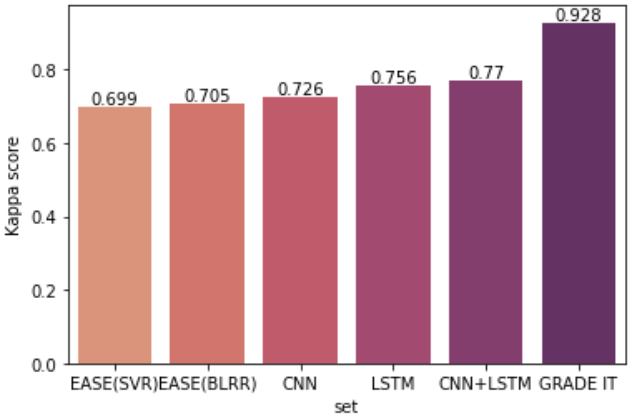

Fig. 5.2- Comparison of different results from different papers

MODEL	SCORE
EASE(SVR)	0.699
EASE(BLRR)	0.705
CNN	0.726
LSTM	0.756
CNN+LSTM	0.761
TSM	0.821

Table 5.1- Comparison of different results from different papers

Developing an AWE that works consistently for all prompts of the essays is vital. BERT proved itself to be the finest when it comes to contextual text embedding and the pretrained BERT $_{base}$ model is vast enough. Handcrafted features were definitely necessary for both scoring and providing mistakes in the output. Provided the right set of features are chosen in the model with right tools to extract those features. The gradeit methodology is definitely a step up when compared to other models, giving us an unparalleled kappa score of 0.928 and is consistent across all types of essays in the data set. This proves that using handcrafted features combined with deep semantic features and encoding with BERT gives us unprecedented results.

Finally, same method is tested for each of the 8 sets of essays separately and the results were consistent and positive. As shown in the figure 5.4 kappa scores were in the range of 0.7-0.8 paving ways for future improvements. ASAP data set has 8 different essay sets as mentioned before each of which not only have a different range of scores but also different set of scores for example set 7 has 4 different scores apart from the total score each of which grading ideas, organization, style and conventions of the writing. Given these wide range of scores we can alter the

model to run the pipeline for every type score of the essay and grade test essays with multiple scores each rating different characteristic of the essay. This also makes the process of giving basic feedback easier for example if an essay has low conventions score grader can print something like “Limited use of conventions of Standard English for grammar, usage, spelling, capitalization, and punctuation for the grade level”.

```
array([[  0,   3,   7,   5,   0,   1,   0,   0,   0,   0],
       [  1,  41, 157,  91,  22,   5,   1,   2,   0,   0],
       [  0,   8,  67,  66,  27,   5,   0,   0,   0,   0],
       [  0,   3,  80, 130,  67,  32,  12,   2,   1,   0],
       [  0,   0,  23,  88,  45,  40,  11,   2,   1,   0],
       [  0,   1,  11,  41,  88, 107,  81,  29,   4,   0],
       [  0,   0,   0,   4,  30,  57,  38,  38,   4,   0],
       [  0,   0,   0,   2,  16,  36,  95, 120,  49,   0],
       [  0,   0,   0,   0,   0,   2,   6,  45,  38,   0],
       [  0,   1,   0,   0,   0,   1,   8,  27,  38,   0]], dtype=int64)
```

Fig. 5.3 Confusion matrix

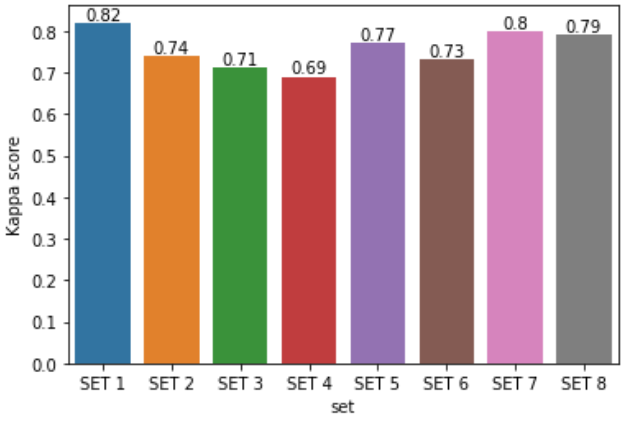

Fig5.4 - Set wise kappa score split

6. CONCLUSION

We recommend the Two-Stage technique (TSM) as it ensures the effectiveness and robustness of AES systems by utilising both hand-crafted and deep encoded features. The prompt relevant score PR and the semantic score Se in the initial step, two different sorts of scores called Pe are calculated. Both of these scores are based on LSTMT. Second, the handcrafted features score is determined. Third, these three scores are combined. The resulting data is input to a boosting tree model, which undergoes additional training and outputs a final score. The results of the studies demonstrate how effective TSM is on the ASAP dataset; our model outperforms numerous trustworthy baselines and performs better than average with a kappa score of about 0.82. In conclusion, both manually created and vector-encoded characteristics are used to provide our system its strength and capabilities.

REFERENCES

[1] [An In-Depth Comparison of 14 Spelling Correction Tools on a Common Benchmark](https://aclanthology.org/2020.lrec-1.228) (Näther, LREC 2020)

[2] T.Chen and C. Guestrin, “Xgboost: A scalable tree boosting system,” CoRR, vol. abs/1603.02754, 2016. [Online]. Available: http://arxiv.org/abs/1603.02754

[3] [A Neural Approach to Automated Essay Scoring](https://aclanthology.org/D16-1193) (Taghipour & Ng, EMNLP 2016)

[4] Alikaniotis, Dimitrios & Yannakoudakis, Helen & Rei, Marek. (2016). Automatic Text Scoring Using Neural Networks. 715-725. 10.18653/v1/P16-1068.

[5] Farag, Youmna & Yannakoudakis, Helen & Briscoe, Ted. (2018). Neural Automated Essay Scoring and Coherence Modeling for Adversarially Crafted Input. 263-271. 10.18653/v1/N18-1024.

[6] Jin, Cancan & He, Ben & Hui, Kai & Sun, Le. (2018). TDNN: A Two-stage Deep Neural Network for Prompt-independent Automated Essay Scoring. 10.18653/v1/P18-1100.

[7] Taghipour, Kaveh & Ng, Hwee. (2016). A Neural Approach to Automated Essay Scoring. 10.18653/v1/D16-1193.

[8] Liu, Jiawei & Xu, Yang. (2019). Automated Essay Scoring based on Two-Stage Learning.

[9] Yannakoudakis, Helen & Briscoe, Ted & Medlock, Ben. (2011). A New Dataset and Method for Automatically Grading ESOL Texts 180-189.

[10] S. Prabhu, K. Akhila and S. S, "A Hybrid Approach Towards Automated Essay Evaluation based on Bert and Feature Engineering," *2022 IEEE 7th International conference for Convergence in Technology (I2CT)*, 2022, pp. 1-4, doi: 10.1109/I2CT54291.2022.9824999.

[11] Z. Zhang and Y. Zhang, "Automated Writing Evaluation System: Tapping its Potential for Learner Engagement," in IEEE Engineering Management Review, vol. 46, no. 3, pp. 29-33, 1 thirdquarter,Sept. 2018, doi: 10.1109/EMR.2018.2866150.

[12] A. Alqahtani and A. Alsaif, "Automatic Evaluation for Arabic Essays: A Rule-Based System," 2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), 2019, pp. 1-7, doi: 10.1109/ISSPIT47144.2019.9001802.

[13] [Should You Fine-Tune BERT for Automated Essay Scoring?](https://aclanthology.org/2020.bea-1.15) (Mayfield & Black, BEA 2020)

[14] P. Phandi, K. Ming A. Chai, and H. Tou Ng, “Flexible domain adaptation for automated essay scoring using correlated linear regression,” 01 2015, pp. 431–439.

[15] Y. Yang, L. Xia and Q. Zhao, "An Automated Grader for Chinese Essay Combining Shallow and Deep Semantic Attributes," in IEEE Access, vol. 7, pp. 176306-176316, 2019, doi: 10.1109/ACCESS.2019.2957582.

[16] Hongbo Chen, Jungang Xu, Ben He, Automated Essay Scoring by Capturing Relative Writing

Quality, *The Computer Journal*, Volume 57, Issue 9, September 2014, Pages 1318–1330, https://doi.org/10.1093/comjnl/bxt117

[17] Burstein, Jill & Kukich, Karen & Wolff, Susanne & Lu, Chi & Chodorows, Martin & Braden-harderss, Lisa & Harrissss, Mary. (2002). Automated Scoring Using A Hybrid Feature Identification Technique. Vol. 1. 10.3115/980451.980879.

[18] Z. Wang, J. Liu and R. Dong, "Intelligent Auto-grading System," 2018 5th IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS), 2018, pp. 430-435, doi: 10.1109/CCIS.2018.8691244.