# A Trait-based Deep Learning Automated Essay Scoring System with Adaptive Feedback

Mohamed A. Hussein[1]
National Center for Examination and Educational Evaluation- NCEEE
Cairo, Egypt

Hesham A. Hassan[2], Mohammad Nassef[3]
Faculty of Computers and Information
Cairo University
Cairo, Egypt

***Abstract*—Numerous Automated Essay Scoring (AES) systems have been developed over the past years. Recent advances in deep learning have shown that applying neural network approaches to AES systems has accomplished state-of-the-art solutions. Most neural-based AES systems assign an overall score to given essays, even if they depend on analytical rubrics/traits. The trait evaluation/scoring helps to identify learners' levels of performance. Besides, providing feedback to learners about their writing performance is as important as assessing their level. Producing adaptive feedback to the learners requires identifying the strengths/weaknesses and the magnitude of influence of each trait. In this paper, we develop a framework that strengthens the validity and enhances the accuracy of a baseline neural-based AES model with respect to traits evaluation/scoring. We extend the model to present a method based on essay traits prediction to give trait-specific adaptive feedback. We explored multiple deep learning models for the automatic essay scoring task, and we performed several analyses to get some indicators from these models. The results show that Long Short-Term Memory (LSTM) based system outperformed the baseline study by 4.6% in terms of quadratic weighted Kappa (QWK). Moreover, the prediction of the traits scores enhance the efficiency of the prediction of the overall score. Our extended model is used in the *iAssistant*, an educational module that provides trait-specific adaptive feedback to learners.**

***Keywords*—AES system; trait evaluation; adaptive feedback; deep learning; neural networks; ASAP**

## I. INTRODUCTION

"Nothing we do to, or for our students is more important than our assessment of their work and the feedback we give them on it [1]." It is widely acknowledged that feedback is a critical element of learning [2]. Both scores and feedback are fundamental aspects of the learning process. Accurate scoring of learners' answers creates a fair way to assess learners' work, which is a very important aspect. However, giving feedback to learners about their answers helps them identify their weaknesses and improve their performance as well.

Rubrics are widely used in evaluating learners' answers to essay questions. Brookhart (2013) defines a rubric as "a coherent set of criteria for learners' work that includes descriptions of levels of performance quality on the criteria [3]." The definition identifies two significant aspects of a good rubric: coherent sets of criteria and descriptions of levels of performance for these criteria. There are two types of rubrics: analytic and holistic rubrics. An analytic rubric evaluates each criterion separately, and a holistic rubric evaluates all criteria simultaneously. Each type has its advantages and disadvantages. Analytic rubrics give formative feedback to learners and are easier to link to instruction. Nevertheless, they take more time to score and achieve acceptable inter-rater reliability than holistic rubrics. Holistic rubrics are faster and suitable for summative assessment (assessment of learning). On the other hand, a single overall score does not communicate information about what to do to improve learning and is not useful for formative assessment (assessment for learning) [4]. It is also interesting to know that research showed that learners prefer AES feedback over peer feedback [5].

Over the past years, various AES systems have been developed to evaluate learners' responses to a given prompt (essay). AES systems automatically asses the quality of the written text and assign a score to each text. The efficiency of these systems depends on the agreement between the human-rater scores and the AES scores[6]. Research in deep learning has led to the development of neural network models for automatic essay scoring task moving away from feature engineering and found that utilizing neural networks to automatic essay scoring task has achieved state-of-the-art outcomes [7]. Utilizing the automatically learned features has added significant benefits to the efficiency of such systems as well [8] [9].

The vast majority of existing Neural based AES systems were developed for holistic scoring to given essays even if they depend on analytical rubrics/traits [10]. The trait evaluation/scoring helps to identify learners' levels of performance. Besides, providing feedback to the learners about their writing requires identifying the strengths/weaknesses and the magnitude of influence of each trait. Based on that, our goal is to develop a framework that strengthens the validity and enhances the accuracy of neural-based AES approaches with respect to traits evaluation/ scoring. Using this framework should help in providing effective adaptive feedback to learners as well.

The following part of the paper is organized as follows: Section 2 describes a brief overview of related work. Section 3 describes the methods and materials, including the AES models (baseline and the augmented), dataset description, training, and testing, in addition to the evaluation metric. Reporting and discussion of results are in Section 4. Then, our conclusion and future improvements are in sections 5.

## II. Related Work

PEG is the earliest AES system that was developed by Ellis Page in 1966. PEG was the starting spark for decades of research into AES. Then, many AES systems have been developed that analyze the quality of text and assign a score to it. AES systems use various manually tuned shallow and deep linguistic features [5].

AES systems can be classified into two main types: i) handcrafted discrete features-based type that is bounded to specific domains, which usually uses natural language processing, latent semantic analysis, or Bayesian network, etc. and ii) automatic feature extraction-based type which usually uses neural networks [5].

Several AES systems include automated scoring alongside providing feedback, e.g., for the first type, Criterion, MY Access, and Writing Pal. Criterion provides an overall score and a learner's feedback using E-rater and Critique as an AES component. Where the E-rater module performs the given essay utomatic scoring task and Critique consists of a set of modules that detect mistakes/errors in mechanics, grammar, and usage. Then, it identifies the issues of discourse and style in writing. MY Access offers instant score and diagnostic feedback based on the IntelliMetric AES system to stimulate the learners to improve their writing ability [8]. Moreover, Writing Pal is classified as an intelligent tutoring system that is mainly concerned with learning tasks and provides the service of evaluating writing tasks with feedback [11]. It targets learners' writing strategies within providing automated feedback. However, it classified as a handcrafted discrete features-based system; the automatic essay scoring model is separate from the feedback part. It uses specific algorithms for each feedback category.

In particular, a few of the other type systems consider scoring the traits and providing the appropriate feedback for each essay. Woods et al. [12] established a new ordinal essay scoring model with extension to use essay traits prediction to give a formative trait-specific feedback to learners. Nevertheless, one of the concerns of their system that their Ordinal Logistic Regression (OLR) model does not perform accurately with large scoring ranges essays (like prompts 1 and 7 in ASAP dataset).

## III. Materials and Methods

### A. Baseline Model

Taghipour and Ng [6], developed an AES system ($AES_{T\&N}$) based on neural networks, which automatically predicts the overall score of a given essay [10]. $AES_{T\&N}$ takes the sequence of words in an essay as input; their model first uses a convolution layer to extract n-gram level features. These features, which capture the local textual dependencies among the words in an n-gram, are then passed to a recurrent layer composed of an LSTM network. It was trained and given state-of-the-art results on the Kaggle's ASAP dataset. The evaluation metric, which is used to evaluate the efficiency of the system, is Quadratic Weighted Kappa (QWK) [6], [8]. They used a 5-fold cross-validation, and for each fold, they distributed the dataset into 60%, 20%, and 20%; training, development, and testing sets, respectively. $AES_{T\&N}$ model architecture is illustrated in Fig. 1.

$AES_{T\&N}$ results show that all model variations (Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Gated Recurrent Units (GRU), and LSTM) succeed to learn the task properly and its performance comparable to or better than the baseline (AES system called 'Enhanced AI Scoring Engine' (EASE)[1]). The authors reported that the LSTM based $AES_{T\&N}$ system outperformed other neural networks (RNN, GRU, and CNN) systems significantly and outperformed the baseline by (4.1%).

$AES_{T\&N}$ system has significantly outperformed the other AES systems, yet there is always an area for improvement to increase the accuracy of scoring. $AES_{T\&N}$ system has predicted only the overall scores, although some of the essays have analytical rubrics/traits. Moreover, it has not provided any feedback to learners.

### B. Proposed Model

Our model ($AES_{AUG}$) is inspired by the baseline model $AES_{T\&N}$ of Taghipour and Ng [6]. We extend and utilize the $AES_{T\&N}$ model to predict not only the overall score for essays but also the traits scores. Besides, we aim to utilize the traits scores to provide adaptive feedback to learners. Fig. 2 presents the $AES_{AUG}$ model architecture, which is described.

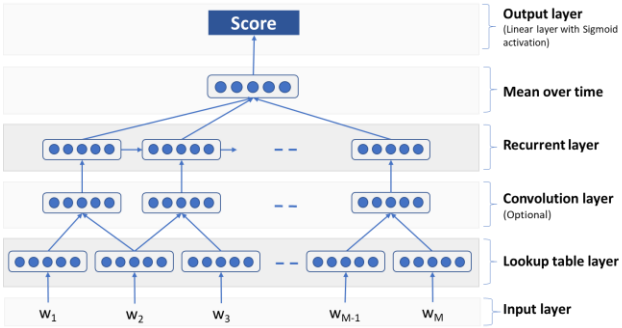


Fig. 1. AEST&N Model Architecture of Taghipour and Ng [6], where the Output Layer Predicts Only the Overall Score.

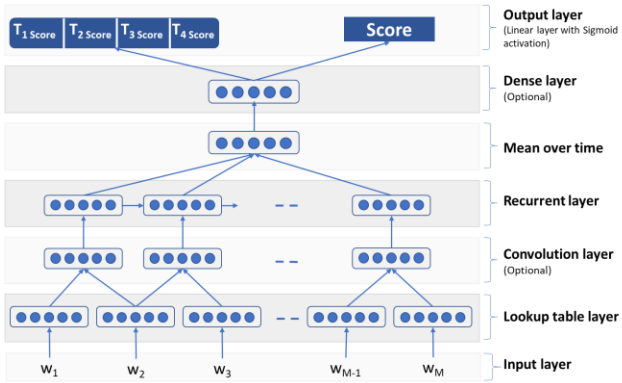


Fig. 2. $AES_{AUG}$ Model Architecture, where the Output Layer Predicts both the Overall Score and Traits' Score.

[1] EASE is an open source handcrafted features-based AES system. It depends on Bayesian linear ridge regression and vector regression techniques. It was the third in the ASAP competition (among 154 systems).

*1) The Lookup Table Layer*; first layer/step of the model transforms each word into dimensional space $d_{LUT}$ . Given a sentence $S = (c_1, c_2, \ldots, c_L)$, the output of the lookup table operation $LUT(S)$ represented in Equation 1.

$$\mathrm{LUT(S)} = (\mathrm{Ec_1}, \mathrm{Ec_2}, \ldots, \mathrm{Ec_L}) \quad (1)$$

where $\mathrm{c_i}$: one-hot representation of the $\mathrm{i-th}$ word in the sentence, and E : is the embedding matrix (learned in the training stage).

*2) The Convolution Layer (optional);* extracts feature vectors from $n-grams$. It can capture local contextual dependencies in writing and, therefore, enhance the efficiency of the system. In order to extract local features from the sequence, the convolution layer applies a linear transformation to all M windows in the given sequence of vectors.

*3) The Recurrent Layer*; processes the input (whether from the convolution layer or directly from the lookup table layer) to generate a representation for the given essay. This representation should encode all the information required for scoring the given essay. Since certain essays are usually long, the proposed model preserved all the intermediate states of the recurrent layer to keep track of the important bits of information. We also experimented with basic RNN vs. GRU vs. LSTM.

In order to control the flow of information during the processing of the input sequence, LSTM units use three gates to discard (forget) or pass the information through time. The following equations formally describe the LSTM function:

$$i_t = \sigma(W_i.x_t + U_i.h_{t-1} + b_i) \quad (2)$$

$$f_t = \sigma\left(W_f.x_t + U_f.h_{t-1} + b_f\right) \quad (3)$$

$$\tilde{c}_t = \tanh(W_c.x_t + U_c.h_{t-1} + b_c) \quad (4)$$

$$c_t = i_t \circ \tilde{c}_t + f_t \circ c_{t-1} \quad (5)$$

$$o_t = \sigma(W_o.x_t + U_o.h_{t-1} + b_o) \quad (6)$$

$$h_t = o_t \circ \tanh(c_t) \quad (7)$$

where $\sigma$: represents the sigmoid function, $\circ$ : denotes multiplication (element-wise), $x_t$ and $h_t$: the input and output vectors at time $t$, respectively, $W_i, W_f, W_c, W_o, U_i, U_f, U_c$ and $U_o$: weight matrices, and $b_i, b_f, b_c$ and $b_o$: bias vectors.

*4) The Mean over Time (MoT);* this layer input is V vectors (the output of the recurrent layer) with variable length, $\mathcal{H} = (h_1, h_2, \ldots, h_V)$,. This layer aggregates these inputs into a fixed-length vector and fed it to the dense layer. Equation 8 describes the function of this layer:

$$MoT(\mathcal{H}) = \frac{1}{M}\sum_{t=1}^{V} h_t \quad (8)$$

*5) The Dense layer (optional);* gives more depth and enhances the efficiency of the model to predict the traits scores in addition to the overall score in the output layer. The mathematical form of the layer is shown in Equation 9:

$$Y = f(W^T X + b) \quad (9)$$

where $W$ is weight matrix (with mini-batch size 32), $b$ is bias vector, $f$ is activations of the previous layer, $X$ is the input of the layer (from MoT layer), and $Y$ is the dense layer output.

*6) The Output layer (Linear Layer with Sigmoid Activation);* maps the dense layer generated output vector to a scalar value. Equation 10 describes applying the sigmoid activation function on the linear layer mapping:

$$s(x) = sigmoid(v.x + b) \quad (10)$$

where: the input vector $(Y)$, $v$ : the weight vector, and $b$ : the bias value. In order to predict the traits scores, we extend the baseline model architecture layers by adding further linear units to the output layer that performs a linear regression to predict traits scores.

We minimized the Mean Squared Error (MSE) between the predicted score and the reference score (human-raters' scores). The $\mathrm{AES_{T\&N}}$ MSE loss function is designed only for the overall score prediction. To fit with predicting the overall and traits scores in our $\mathrm{AES_{AUG}}$ model, we adjusted the $\mathrm{AES_{T\&N}}$ MSE loss function (shown in Equation 11) to compute the overall loss function as a linear combination of multi loss functions (shown in Equation 12), back-propagating the error gradients to the embedding matrix.

$$MSE\ (s, s^*) = \frac{1}{N}\sum_{i=1}^{N}(\mathcal{S}_i - \mathcal{S}_i^*)^2 \quad (11)$$

$$MSE\ (s, s^*) = \frac{1}{N}\left(\sum_{i=1}^{N}(\mathcal{S}_i - \mathcal{S}_i^*)^2 + \sum_{j=1}^{T}\sum_{i=1}^{N}(t_{ij} - t_{ij}^*)^2\right) \quad (12)$$

where $T$: a number of a specific prompt traits, given $N$: number of training essays and their corresponding normalized reference overall scores $\mathcal{S}_i^*$ , and $t_{ij}^*$ : traits normalized reference scores. The model computes the predicted overall scores $\mathcal{S}_i$ and traits scores $t_{ij}$ for all training essays.

### C. Dataset

AES research has been dominated for the last eight years by the dataset from the 2012 Automated Student Assessment Prize (ASAP) competition [13]. It was established by Kaggle and funded by the Hewlett Foundation. ASAP competition has provided the data and all the required information (hand-crafted features), which can help to evaluate AES systems that use machine learning algorithms. ASAP consists of 12.976 essays, with average length 150-to-550 words per essay, each double scored (Cohen's = 0.86) [8]. The dataset consists of eight tasks/prompts; each task is an essay that has learners' responses. ASAP provided the scoring guides, raters' exemplars, and practice sets for each task. Five tasks employed a holistic scoring rubric, one was scored with a two-trait analytic rubric, and two were scored with a multi-trait analytic rubric but reported as an overall score [14]. Shermis [15] provides a summary of the competition, and most of the recent papers report their results using the same public dataset [16][6][12][17][18][6][19].

In this research, we have used the ASAP data and specifically task 7 data. Task 7 was selected because it has a multi-trait analytic rubric that can be used for formative feedback to learners, and it has the largest dataset (1,569

essays) on the multi-trait analytic rubric-based tasks. The type of writing in task 7 is persuasive/narrative/expository. The prompt asks learners to write a story about patience. The scoring rubric has four traits: ideas, organization, style, and conventions. Each trait score ranges from 0-3. Each score in each trait has a description that guides the rater to identify the appropriate score (level) to each text. In ideas, for example, if the ideas are clearly focused on the topic and are thoroughly developed with specific relevant details, a score of 3 should be assigned. If the ideas are somewhat focused on the topic and are developed with a mix of specific and/or general details, a score of 2 should be assigned. If the ideas are minimally focused on the topic and developed with limited and/or general details, a score of 1 should be assigned. If the ideas are not focused on the task and/or are undeveloped, a score of 0 should be assigned. For objectivity and accuracy, two raters should score the response of each learner for each trait. Then, the scores were summed independently for Rater1 and Rater2 to form the resolved score (0-30) by adding the sum of the two raters.

### D. Training and Testing

We have followed the dataset split by Taghipour and Ng [6], so we used a 5-fold cross-validation model to assess our proposed system. Data, in each fold, is distributed into 60%, 20%, and 20%; training, development, and test sets, respectively. For prompt no. 7 and each of its four traits, the fold predictions have been aggregated and evaluated together. In order to evaluate the system efficiency, the results are averaged across the four traits. See Fig. 3. The essays have been tokenized by the NLTK[2] tokenizer that lowercases the letters and normalizes the reference scores to the range of [0, 1]. For the system performance evaluation, we rescaled the system-predicted normalized scores to the original range of scores.

In some experimental scenarios, we used a different split ratio in each fold to maximize the training data size for the best training: 80% of the data as a training set, and 20% as the test set.

We followed the $AES_{T\&N}$ by using the RMSProp optimization algorithm [20] to minimize the MSE loss function over the training data. We also used dropout regularization to avoid overfitting. If the norm of the gradient is larger than a threshold, it will be clipped. We did not use any early stopping method. We trained the model for a fixed 50 epochs, and after each epoch, we monitored the model efficiency on the development set.

The system hyper-parameters are several: To train the network, we have used RMSProp optimizer with the decay rate ($\rho$) set to 0.9. We used pre-trained word embeddings[3], released by Zou et al. [21] to initialize the lookup table layer. The hyper-parameter settings are listed in Table I. We used Nvidia GEFORCE GTX 1050 GPU to perform our experiments in parallel.

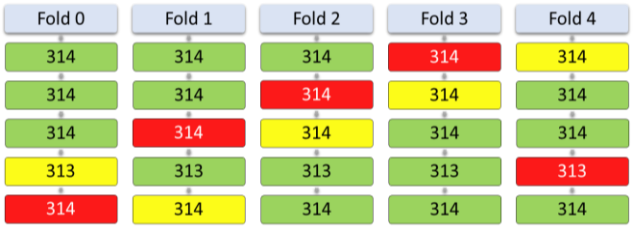


Fig. 3. Prompt no. 7 Dataset Folds Distribution, Green is Training, Yellow is Validation, and Red is Test Set.

TABLE I. $AES_{AUG}$ MODEL HYPER-PARAMETERS

| Parameter | Parameter meaning/description | Value |
|---|---|---|
| $d_{LT}$ | Word embedding dimension | 50 |
| $d_r$ | Output dimension of the recurrent layer | 300 |
| $l$ | Word context window size | 3 |
| $d_c$ | Word convolution units | 50 |
| *drop-rate* | Dropout probability | 0.5 |
| *batch-size* | Mini-batch size[a] | 32 |
| *Learn-rate* | Base learning rate | 0.001 |

[a]a fixed 50 epochs.

### E. Evaluation

The evaluation of AES systems is always done by comparing the AES scores to the scores assigned by human raters. Various statistics tests of correlation or agreement are used for this purpose, including Pearson's correlation, Spearman's correlation, and QWK [22]. QWK was identified as the official evaluation metric for ASAP. In this paper, we used the QWK to evaluate our system to the well-established baseline ($AES_{T\&N}$) that used the same dataset. The QWK is a commonly used measure of the degree of agreement among raters (a.k.a. inter-rater reliability). The following part illustrates how QWK is computed.

A weight matrix $W$ is created based on Equation 13:

$$W_{i,j} = \frac{(i-j)^2}{(N-1)^2} \quad (13)$$

where $i$ and $j$ are the reference scores, and the hypothesis scores (AES scores), respectively. $N$ refers to the number of all possible scores. $O$ is a matrix calculating like $O_{i,j}$ refers to the number of texts which are given a score $i$ by the rater and an AES score $j$. A count matrix $E$ is computed to represent the outer product of histogram vectors of the two scores. The sum of elements in $O$ is equal to the sum of elements in $E$ as the matrix $E$ is normalized. Lastly, based on matrices $O$ and $E$, the QWK is computed as of Equation 14:

$$k = 1 - \frac{\sum_{i,j} W_{i,j} O_{i,j}}{\sum_{i,j} W_{i,j} E_{i,j}} \quad (14)$$

Our comparison between the $AES_{AUG}$ and $AES_{T\&N}$ is always by using the QWK values. A one-tailed paired t-test is always used to check the significance of the differences between the two systems.

[2] http://www.nltk.org
[3] http://ai.stanford.edu/~wzou/mt

## IV. Results and Discussion

We describe in this part our experiments and results. In the case of overall scores, we mention the results and then evaluate our system to the baseline system ($AES_{T\&N}$). In the case of traits scores, we present only the results of our $AES_{AUG}$ system, and its QWK evaluation as the $AES_{T\&N}$ system did not predict traits scores.

We started our experiments by replicating the $AES_{T\&N}$[4] model results over the ASAP dataset. Taghipour and Ng [6] (using $AES_{T\&N}$) experimented and explored a variety of neural network model architectures like CNN, basic RNN, GRU, and LSTM without using an MoT layer. After replicating the $AES_{T\&N}$ systems (CNN, RNN, GRU, and LSTM) and producing the same QWKs results, we extended the model to the $AES_{AUG}$ model architecture. We trained the model with the training data (described in section 2.3), including the overall score and the four traits reference-scores (by 2 human raters as described in section 2.3). We started by simulating the human approach in scoring traits that every rater gives a score, and the trait score is the summation of the two raters' scores, so $AES_{AUG}$ systems predicted two scores for every trait, and we summed them. We got the same QWK (0.805) for the overall score (on Fold 4) and QWK [0.715, 0.623, 0.581, 0.443] for the first predicted traits scores and [0.723, 0.656, 0.568, 0.476] for the second predicted traits scores, with an average [0.598][5].

We found that the predicted traits scores have low QWK values, so we analyzed the case by calculating the QWK among the first human rater (H-R1), the second human rater (H-R2), and each of $AES_{AUG}$ predicted scores (A-R3 & A-R4). Table II shows QWK for traits scores of the human raters and the $AES_{AUG}$ system (using the best model, which is LSTM). We noticed that the agreement (QWK) between the human raters (0.64) is lower than the agreement (QWK) between any $AES_{AUG}$ prediction and any of the human raters (0.66, 0.67, 0.68 and 0.68); All the QWKs are shown in Table II. In our attempt to understand the logic behind this low agreement, we examined the prompt content and rubrics with the help of two English language specialists. They confirmed that the definitions of the level descriptors in the rubrics are not clear and definite, which may lead to different interpretations between raters, which accordingly may lead to a low agreement between raters. They also added that using the summation of the two raters on each trait (as described on the ASAP scoring guide) will provide a more accurate and objective indicator for a learner's performance.

In order to enhance the traits QWK scores for $AES_{AUG}$ systems, we changed our score calculation approach, i.e., before training the system, we calculated one score for each trait by summing the two human scores. Then, we calculated the QWK score for each trait between one reference-score and one $AES_{AUG}$ system predicted score. As a result of that change in score calculation methods, we got higher QWKs for the traits scores [0.820, 0.767, 0.767, 0.733], respectively, with an average QWK of [0.771]. We also noticed that the traits scores prediction within $AES_{AUG}$ model architecture enhanced the accuracy of predicting the overall score [0.851] (on Fold 2) to outperform the baseline $AES_{T\&N}$ best model (LSTM) which was [0.805] with 4.6% improvement. It even outperformed the best result for prompt no. 7, which is LSTM ensembles (10 runs), which QWK was [0.811] with a 4% improvement. As shown in Table III, predicting traits scores always leads to improvement in the $AES_{AUG}$ overall score.

Table III shows the QWKs of our $AES_{AUG}$ models on prompt no. 7 overall score and four traits scores. It also shows the $AES_{T\&N}$ systems replicated results for the overall score. The statistical significance of improvements is marked with '*'.

We produced the $AES_{AUG}$ systems for all models (CNN, RNN, GRU, and LSTM)[6]; all results are shown in Table III. Based on Table III, all models can predict the overall and traits scores competitively compared to the baseline. However, we agree with Taghipour and Ng [6] findings that LSTM has performed better than the other models significantly, and it has outperformed the baseline model by (4.6%). Nevertheless, the least accurate model is basic RNN, which does not work precisely as GRU or LSTM. Such a finding can be due to the moderately long sequences of words in texts. Both LSTM and GRU demonstrate efficient learning of long-term dependencies and sequences. Therefore, we believe this is of the RNN's poor performance points. The CNN model is the fastest in the training and the evaluation compared to other models.

We further investigated the overall and traits scores predicted by our best model ($AES_{AUG}$ LSMT), for the predicted and original in ASAP dataset. We presented the results in Fig. 4((a) for overall score, (b), (c), (d), and (e) for the traits). The graphs show the system predictions are less varied and positively contribute to the performance of our proposed approach.

TABLE II. QWK among Human Raters (H-R1 & H-R2) and each of $AES_{AUG}$ Predicted Scores (A-R3 & A-R4)

| **Raters** | **Average QWK score** |
|---|---|
| H-R1 vs. H-R2 | 0.641 |
| A-R3 vs. A-R4 | 0.906 |
| H-R1 vs. A-R3 | 0.684 |
| H-R1 vs. A-R4 | 0.680 |
| H-R2 vs. A-R3 | 0.669 |
| H-R2 vs. A-R4 | 0.670 |

TABLE III. The QWK of the $AES_{AUG}$ Several Neural Network Models and the $AES_{T\&N}$

| Systems | $AES_{T\&N}$ QWK | $AES_{AUG}$ QWK | $AES_{AUG}$ traits QWK | | | | |
|---|---|---|---|---|---|---|---|
| | | | *1* | *2* | *3* | *4* | *average* |
| **CNN** | 0.746 | **0.822*** | 0.793 | 0.717 | 0.714 | 0.700 | 0.731 |
| **RNN** | 0.743 | **0.760*** | 0.733 | 0.656 | 0.652 | 0.641 | 0.671 |
| **GRU** | 0.752 | **0.827*** | 0.837 | 0.749 | 0.728 | 0.700 | 0.754 |
| **LSTM** | 0.805 | **0.851*** | 0.820 | 0.767 | 0.767 | 0.733 | **0.771** |

* $p < .0001$.

[4] https://github.com/nusnlp/nea

[5] We tried to add L2 regularization and 256 dense layers, but the model extracted was not better than the one that was concluded.

[6] All the mentioned neural network models are unidirectional and include the MoT layer. The convolution layer is included in the CNN model only.

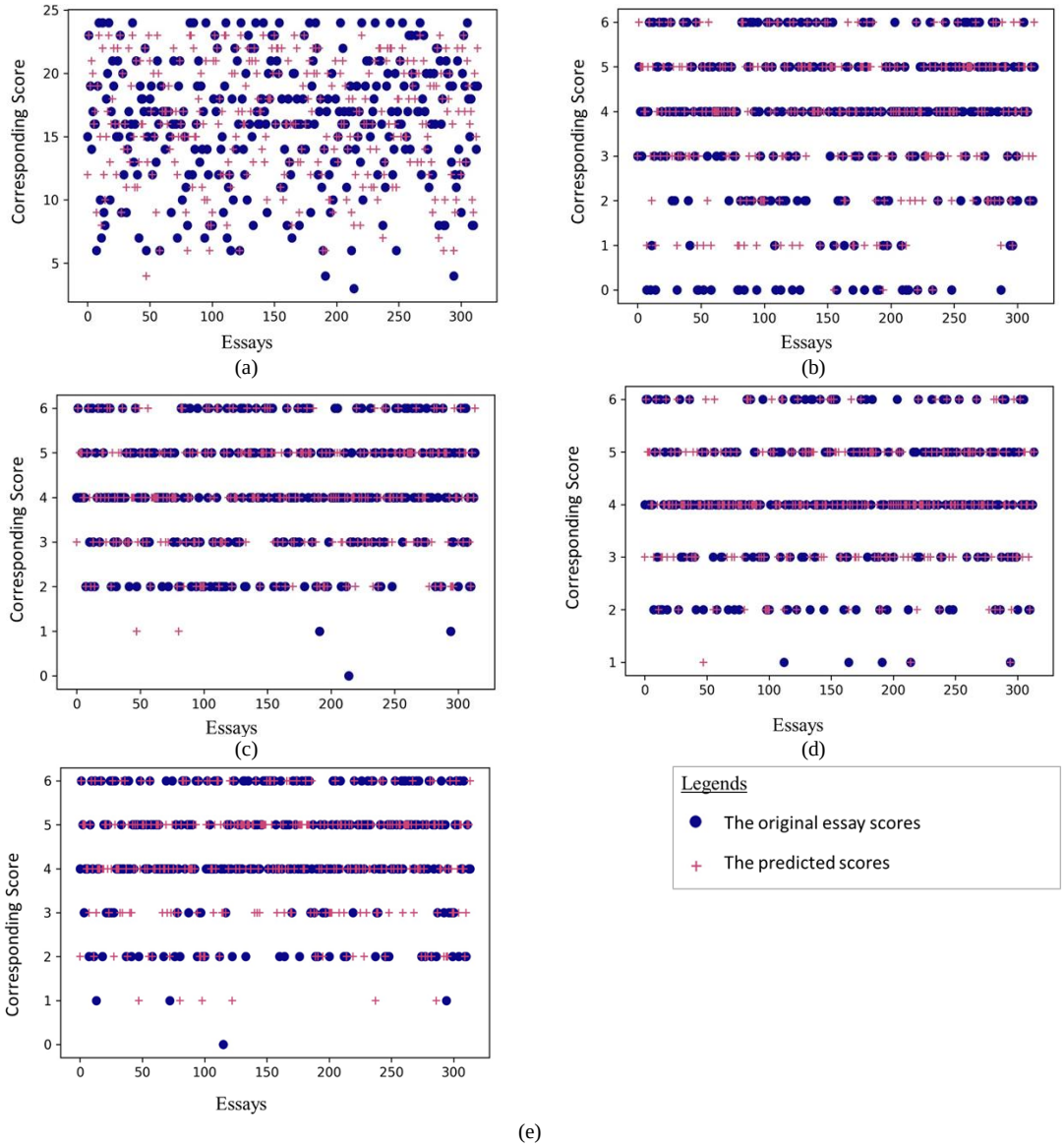


Fig. 4. The Graphs show for Prompt no. 7 and its Traits, the System Predictions are Less Varied and Positively Contribute to the Efficiency of our Proposed Approach. (a) Representing the Overall Score, While (b), (c), (d), and (e) Represent the Four Traits' score, respectively. The Blue Circles Represent the Original Essay Scores, and the Red Pluses the Predicted Scores. All Predicted Scores are Mapped to their Original Scoring Scale.

In the end, we experimented with using a different split to the dataset from the one described in Section III-D (which is 60% training, 20% validation, and 20% testing). Thus, we merged the training set with the validation set to be 80% training and 20% testing. It has achieved better QWK scores for the overall score to be [0.858] instead of [0.851], which means that the availability of a bigger training set will improve the results.

Finally, we used the above method, and its results in the iAssistant, an educational module that provides trait-specific adaptive feedback to learners. As shown in Fig. 5, iAssistant provides learners with predicted scores on multiple rubric traits and levels of performance per each trait. In addition to that, it helps learners to evaluate the length of their essay on a scale of 3 levels (short, good, and long).

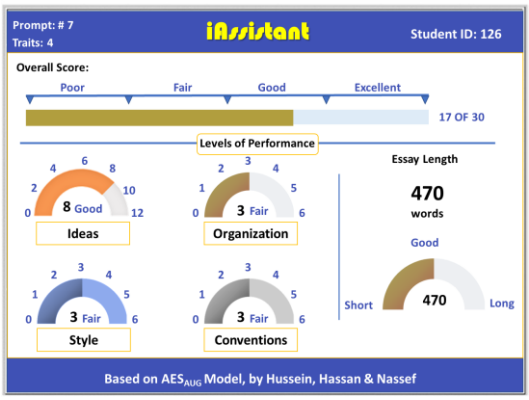


Fig. 5. An Example of iAssistant in use: Predicted Scores on Multiple Rubric Traits and Levels of Performance. In Addition to Representing the Overall Score and Length of the Essay.

## V. Conclusions

In this paper, we have proposed a framework, based on deep learning models that strengthens the validity and enhances the accuracy of a baseline system with respect to traits' evaluation/scoring. Our method does not rely only on overall score prediction but also on essay traits prediction to give trait-specific adaptive feedback. We explored multiple deep learning models for the automatic essay scoring task.

Based on our experiments, we can conclude that our proposed $AES_{AUG}$ model outperformed all the previously used AES models (CNN, RNN, GRU, and LSTM). Including traits in training has significantly improved the learning process. Thus, our $AES_{AUG}$ system has significantly increased the accuracy of the overall and traits scores for essays using analytic-rubrics. This point highlights the contributions of our model over all the previous models.

It is also found that the $LSTM_{AUG}$ model, like the $AES_{T\&N}$ system, proves to be the best model to predict scores for essays that include relatively long sequences of words which is consistent with the nature of the LSTM models. However, adding a dense layer between the MoT layer and the output layer did not improve the results of our $AES_{AUG}$ model. We can also assume, based on our experiments, that increasing the training data has a positive effect on the accuracy of $AES_{AUG}$ scores.

Additionally, it is very important to note that the clarity of the definition of the scoring rubrics strongly influences the accuracy of both human and $AES_{AUG}$ scores, which accordingly affects the quality of the adaptive feedback that can be given to the learners. In other words, the more the rubric is clear and definite, the more the $AES_{AUG}$ scores are accurate, and the feedback is more specific.

Finally, our proposed $AES_{AUG}$ model offers a new methodology that may be interesting to the users, and it provides more accurate results without requiring a high configuration of hardware.

## VI. Future Work

The future directions of this work may be to highlight the words and sentences that made the AES system give a specific score for further analysis and adaptive feedbacking, in addition to training and testing the model on a larger dataset with well-defined rubrics.

## References

[1] S. Brown, 500 Tips on Assessment. Routledge, 2004.

[2] T. C. Stephen, "Using Automated Essay Scoring to Assess Higher-Level Thinking Skills in Nursing Education," 2019.

[3] S. M. Brookhart, How To Create and Use Rubrics. Ascd, 2013.

[4] A. J. N. and S. M. Brookhart, Educational assessment of students. Pearson Merrill Prentice Hall, 2007.

[5] M. Lu, Q. Deng, and M. Yang, "EFL writing assessment: Peer assessment vs. automated essay scoring," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2020.

[6] K. Taghipour and H. T. Ng, "A Neural Approach to Automated Essay Scoring," in Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016, pp. 1882–1891.

[7] F. Nadeem, H. Nguyen, Y. Liu, and M. Ostendorf, "Automated Essay Scoring with Discourse-Aware Neural Models," 2019.

[8] M. A. Hussein, H. Hassan, and M. Nassef, "Automated language essay scoring systems: A literature review," PeerJ Comput. Sci., vol. 2019, no. 8, 2019.

[9] N. W. Solomons et al., "[Use of tests based on the analysis of expired air in nutritional studies].," Arch. Latinoam. Nutr., vol. 28, no. 3, pp. 301–17, 1978.

[10] Z. Ke and V. Ng, "Automated Essay Scoring: A Survey of the State of the Art," 2019.

[11] R. D. Varner, L. K. Crossley, and S. A. Mcnamara, "Developing pedagogically-guided algorithms for intelligent writing feedback," 2013.

[12] B. Woods, D. Adamson, S. Miel, and E. Mayfield, "Formative Essay Feedback Using Predictive Scoring Models," dl.acm.org, vol. Part F1296, pp. 2071–2080, Aug. 2017.

[13] S. Dikli, "Automated essay scoring," Turkish Online Journal of Distance Education, vol. 7, no. 1. pp. 45–56, 2006.

[14] M. Shermis and B. H. Education, "Contrasting state-of-the-art automated scoring of essays: Analysis," Annual national council on measurement in education meeting. 2012.

[15] M. D. Shermis, "State-of-the-art automated essay scoring: Competition, results, and future directions from a United States demonstration," Assess. Writ., vol. 20, pp. 53–76, 2014.

[16] T. Dasgupta, A. Naskar, R. Saha, and L. Dey, "Augmenting Textual Qualitative Features in Deep Convolution Recurrent Neural Network for Automatic Essay Scoring," aclweb.org, pp. 93–102, 2018.

[17] F. Dong and Y. Zhang, "Automatic Features for Essay Scoring-An Empirical Study," 2016.

[18] P. Phandi, K. A. Ming Chai, and H. Tou Ng, "Flexible Domain Adaptation for Automated Essay Scoring Using Correlated Linear Regression," Association for Computational Linguistics, 2015.

[19] D. Alikaniotis, H. Yannakoudakis, and M. Rei, "Automatic Text Scoring Using Neural Networks," 2016.

[20] Y. N. Dauphin, H. De Vries, and Y. Bengio, "Equilibrated adaptive learning rates for non-convex optimization," in Advances in Neural Information Processing Systems, 2015, vol. 2015-Janua, pp. 1504–1512.

[21] W. Y. Zou, R. Socher, D. Cer, and C. D. Manning, "Bilingual Word Embeddings for Phrase-Based Machine Translation," Association for Computational Linguistics, 2013.

[22] H. Yannakoudakis and R. Cummins, "Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications," 2015.