Automated Cross-prompt Scoring of Essay Traits

Robert Ridley, Liang He, Xin-yu Dai,* Shujian Huang, Jiajun Chen
National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, 210023, China
Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing, 210023, China
{robertr, heliang}@smail.nju.edu.cn, {daixinyu, huangsj, chenjj}@nju.edu.cn

Abstract

The majority of current research in Automated Essay Scoring (AES) focuses on prompt-specific scoring of either the overall quality of an essay or the quality with regards to certain traits. In real-world applications obtaining labelled data for a target essay prompt is often expensive or unfeasible, requiring the AES system to be able to perform well when predicting scores for essays from unseen prompts. As a result, some recent research has been dedicated to cross-prompt AES. However, this line of research has thus far only been concerned with holistic, overall scoring, with no exploration into the scoring of different traits. As users of AES systems often require feedback with regards to different aspects of their writing, trait scoring is a necessary component of an effective AES system. Therefore, to address this need, we introduce a new task named Automated Cross-prompt Scoring of Essay Traits, which requires the model to be trained solely on non-target-prompt essays and to predict the holistic, overall score as well as scores for a number of specific traits for target-prompt essays. This task challenges the model's ability to generalize in order to score essays from a novel domain as well as its ability to represent the quality of essays from multiple different aspects. In addition, we introduce a new, innovative approach which builds on top of a state-of-the-art method for cross-prompt AES. Our method utilizes a trait-attention mechanism and a multi-task architecture that leverages the relationships between each trait to simultaneously predict the overall score and the score of each individual trait. We conduct extensive experiments on the widely used ASAP and ASAP++ datasets and demonstrate that our approach is able to outperform leading prompt-specific trait scoring and cross-prompt AES methods.

Introduction

Automated Essay Scoring (AES) is the task of using computation to score an essay according to either its overall quality or its quality with regards to certain traits, e.g., organization, prompt adherence, narrativity, etc. The majority of AES research focuses on scoring in a prompt-specific setting, whereby the model is both trained and tested on essays belonging to the same prompt. This can be seen in the top-left and bottom-left images of Figure 1, where both the training and test data are drawn from the same distribution D_A.

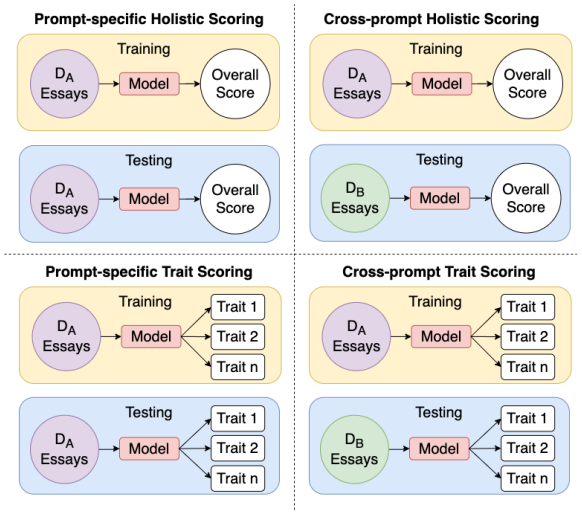

Figure 1: AES Tasks Summary

Early work in prompt-specific AES primarily leveraged handcrafted features used in combination with regression, classification or ranking algorithms to award a holistic, overall score (Burstein et al. 1998; Miltsakaki and Kukich 2004; Rudner and Liang 2002; Farra, Somasundaran, and Burstein 2015; Chen and He 2013).

Following the rise of deep learning, a number of neural-network-based approaches have also been effective for holistic prompt-specific AES (Alikaniotis, Yannakoudakis, and Rei 2016; Taghipour and Ng 2016; Dong and Zhang 2016; Dong, Zhang, and Yang 2017; Tay et al. 2018).

In consideration for the need for richer feedback in AES systems, some research has been dedicated to scoring essays with regards to different traits. For example, Persing, Davis, and Ng (2010) and Wachsmuth, Al-Khatib, and Stein (2016) model the quality of the organization; Ke et al. (2018) score essays with regards to their persuasiveness; Persing and Ng (2013) model thesis clarity; and Persing and Ng (2014) provide feedback with regards to prompt adherence. More recently, researchers have explored providing feedback for both the overall quality and the quality across multiple traits

*Corresponding author

(Mathias and Bhattacharyya 2018, 2020; Hussein, Hassan, and Nassef 2020).

As researchers have previously pointed out (Jin et al. 2018; Ridley et al. 2020), it is often the case that real-world AES systems do not have access to ample target-prompt essays, so it is necessary to develop approaches that are effective when predicting scores for essays belonging to prompts not present in the training data. As a result, some recent research has explored cross-prompt AES, where a model is trained on either non-target-prompt essays in conjunction with low quantities of target-prompt essays (Phandi, Chai, and Ng 2015; Cummins, Zhang, and Briscoe 2016) or only on non-target-prompt essays (Attali, Bridgeman, and Trapani 2010; Jin et al. 2018; Ridley et al. 2020) to achieve better generalized approaches that can perform better for essays belonging to novel prompts. This task is visualized in the top-right image of Figure 1, whereby the training data and test data are drawn from different distributions, D_A and D_B.

Cross-prompt AES research has so far only been concerned with grading essays according to their holistic, overall quality, with no existing research exploring trait scoring in this setting. Due to the need for high performance in the cross-prompt setting *and* the need for an ability to provide feedback across different traits, we contest that an effective AES system should incorporate these two components. Therefore, we introduce a new AES task named Automated Cross-prompt Scoring of Essay Traits, whereby the model is required to be trained solely on non-target-prompt essays and to predict the overall score as well as scores for a number of traits for essays belonging to the target prompt. This is visualized in the bottom-right image of Figure 1, where the training and test data are drawn from different distributions and the output is the score for different traits.

This new task exhibits two main challenges: first, the model needs to possess sufficient generalizability in order to perform well on novel domains; second, the model needs to be able to represent essay quality from different aspects to be effective in scoring various essay traits.

In approaching the task of Automated Cross-prompt Scoring of Essay Traits, we address two issues: first, partial coverage of scores for traits across different prompts leads to insufficient training data for certain traits. For example, it may be the case that essays from only two prompts have scores for the *narrativity* trait. As a result, a model trained to award a score for this trait can be trained only on the essays from these two prompts. Second, there is a high degree of relatedness between different traits. For example, an essay that scores well on the *word choice* dimension can also be expected to score well on the *conventions* dimension.

To address the issue of limited training data that arises from partial trait coverage, we introduce a multi-task approach named Cross-prompt Trait Scorer (CTS)[1] that simultaneously predicts the overall score and the scores for all traits. This enables the model to train on all data in the training set to learn a more robust representation. In order to address the issue of inter-trait relationships, we design a trait-attention mechanism, which learns to utilize the most relevant trait information in predicting the score for each trait.

In summary, the contributions of this work are as follows:

- We introduce a new task named Automated Cross-prompt Scoring of Essay Traits, which integrates cross-prompt essay scoring and essay trait scoring, two vital tasks in real-world AES solutions.
- We design a novel approach named Cross-prompt Trait Scorer (CTS), which utilises a multi-task approach to address the issue of limited training data due to partial trait coverage.
- We design a trait-attention mechanism to take advantage of the relationships that exist between the different traits.

Related Work

Prompt-specific Holistic Scoring

Initial research in prompt-specific holistic essay scoring primarily utilized handcrafted features combined with traditional machine learning algorithms. For instance, Rudner and Liang (2002) explore using various word, phrase and argument features in combination with two Bayesian-based models; Chen and He (2013) use lexical, syntactic, grammar and prompt-specific features in combination with a ranking-based approach; and Miltsakaki and Kukich (2004) use coherence features in combination with the *e-rater* scoring system (Burstein et al. 1998).

In recent years, neural-network-based approaches have proven effective. The earliest such approach (Taghipour and Ng 2016) uses one-hot encodings in combination with convolutional and recurrent layers. Other researchers have since developed more sophisticated architectures, with Alikaniotis, Yannakoudakis, and Rei (2016) learning score-specific word embeddings, Dong and Zhang (2016) representing essays at both word-level and sentence-level in a hierarchical approach, Dong, Zhang, and Yang (2017) using an attention-pooling mechanism to attend to the most relevant parts of the text, and Tay et al. (2018) measuring similarity across different sections of text to model coherence.

Prompt-specific Trait Scoring

For essays graded with the use of a rubric, providing feedback with regards to different traits can be of great use to users. In the prompt-specific setting, a number of works have explored automated scoring of different traits. For instance, Persing, Davis, and Ng (2010) present a new annotated corpus and utilize sequence-alignment, alignment-kernel and string-kernel techniques to score essays according to the quality of their organization; Wachsmuth, Al-Khatib, and Stein (2016) also score essays in the organization dimension, which they perform through argument mining; and Ke et al. (2018) address modeling argument persuasiveness and the attributes of those arguments in student essays.

More recently, some works have explored methods to award holistic overall scores as well as scores for multiple attributes. Mathias and Bhattacharyya (2018) introduce a new multi-trait dataset and use a random forest classifier;

[1] Our code is available at https://github.com/robert1ridley/cross-prompt-trait-scoring.

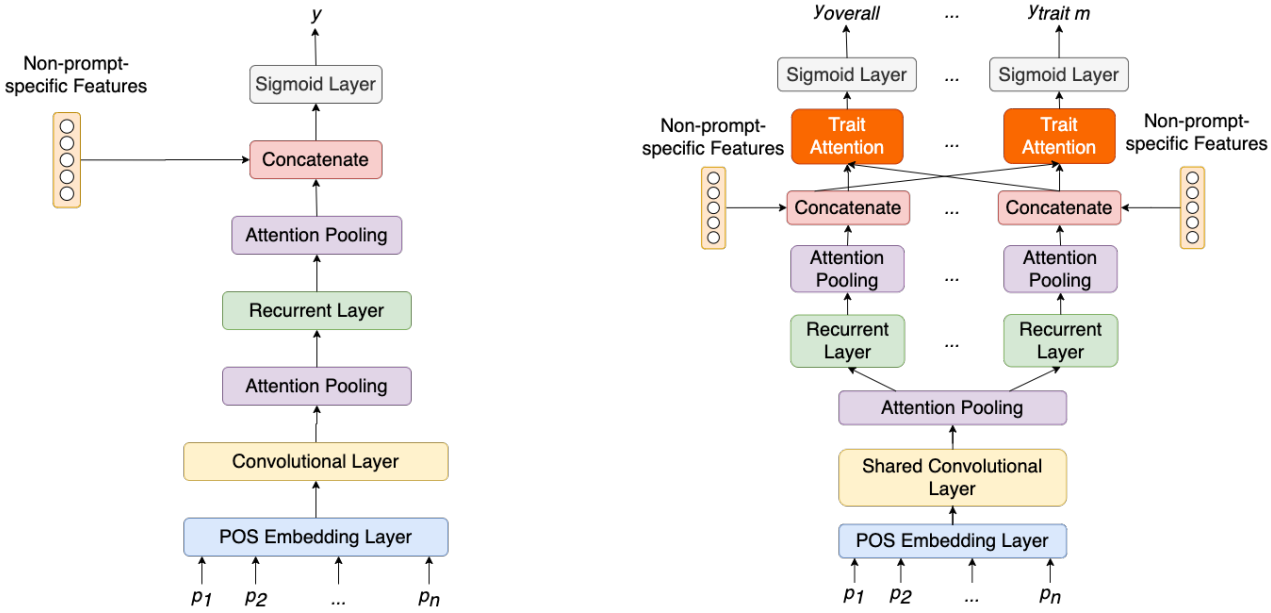

Figure 2: Architecture of both PAES (Ridley et al. 2020) on the left and our proposed CTS model (right)

Mathias and Bhattacharyya (2020) adapt some leading approaches for prompt-specific holistic scoring to the task of trait scoring; and Hussein, Hassan, and Nassef (2020) also adapt a leading prompt-specific holistic scoring method, employing a multi-task architecture to output the overall score and scores for various traits simultaneously.

Cross-prompt Holistic Scoring

Obtaining a sufficient quantity of pre-graded essays to a specific prompt is expensive and often not possible. Therefore, it is necessary for AES systems to be able to perform well when trained on essays belonging to non-target prompts. This necessity has given rise to research into cross-prompt holistic essay scoring. Attali, Bridgeman, and Trapani (2010) explore the use of non-content features with a regression model. Other early works in this setting (Phandi, Chai, and Ng 2015; Cummins, Zhang, and Briscoe 2016) train models with large quantities of non-target-prompt and a small quantity of target-prompt essays and perform transfer learning to score target-prompt essays.

Some more recent research has explored cross-prompt holistic scoring in the scenario where there is no access to target-prompt essays. Jin et al. (2018) apply a two-stage approach, in which the first stage utilizes prompt-independent features to award pseudo labels to target-prompt essays. In the second stage, the pseudo-labeled essays are then used as training data in a neural network which awards the final scores. Ridley et al. (2020) apply a single-stage neural-network-based method that utilizes a set of general features to award scores to target-prompt essays.

One drawback of current research into cross-prompt essay scoring is that only scoring of the holistic dimension is explored. Since feedback regarding different traits is also of great value to users, we believe it is necessary to provide scores for different traits in the cross-prompt setting. As a result, we put forward a new task named Automated Cross-prompt Scoring of Essay Traits, which we introduce in the next section.

Task Definition

In Automated Cross-prompt Scoring of Essay Traits, all training samples are non-target-prompt essays and belong to the source domain $D_A = \{x_A^i, Y_A^i\}_{i=1}^{N_A}$, where N_A is the number of essays in D_A. For testing, essays belong to the target prompt and are within the target domain $D_B = \{x_B^j, Y_B^j\}_{j=1}^{N_B}$, with N_B being the number of essays in D_B. Traits for each essay belong to a possible trait set $Y = \{y_1, y_2, ..., y_M\}$, where y_1 is the overall score. Each essay in the training set possesses gold-label scores for a trait subset $Y_A^i \subset Y$. Essays in D_B all belong to the same prompt, so $Y_B \subset Y$, and $Y_B^j = Y_B$ for every j-th essay in D_B.

Approach

Our approach builds on top of PAES (Ridley et al. 2020), a leading method in cross-prompt AES. The architecture for this method is displayed in the image on the left-hand side of Figure 2. This approach uses part-of-speech (POS) embeddings to learn generalized syntactic representations. First, a convolutional layer is applied to each sentence and attention pooling is applied to achieve sentence-level representations. Then, these representations are fed through a recurrent layer implemented with an LSTM (Hochreiter and Schmidhuber 1997) followed by a second attention pooling layer to learn the full essay representation. A set of non-prompt specific features are then concatenated with the essay representation before a linear layer with a sigmoid activation is applied to predict a single score.

This approach can be directly applied to our proposed task in a naive fashion, whereby the model is trained on each trait independently. This method, however, suffers from two shortcomings. First, if a target trait is underrepresented in the training data, then there will be insufficient data to train a robust model. Second, the traits are not independent of each other, but are in fact related. This naive approach does not utilize any inter-trait relationships that exist.

To address these shortcomings, we design a model named Cross-prompt Trait Scorer (CTS), which is depicted in the image on the right-hand side of Figure 2. In consideration for the issue of insufficient data due to partial trait coverage, our design applies a multi-task-based architecture. This enables the model to train on all samples in the dataset in order to learn a more robust encoder representation. To address the inter-trait relationships issue, we implement shared layers at the lower levels of the architecture followed by private layers at the higher levels. The shared low-level layers aim to learn low-level representations that are useful across all tasks. Higher-level layers in multi-task architectures have been shown to represent more complex information (Sanh, Wolf, and Ruder 2019). Thus high-level private layers are employed to learn more task-specific representations. In addition, to share information between traits more explicitly, we design a trait-attention mechanism, which allows each trait to focus on the relevant information from each of the other traits. Details of our design are introduced in the following sections.

Shared Layers

The parameters in the lower layers of our model are shared to enable the sharing of information relevant to all tasks. As with Ridley et al. (2020), we first perform POS-tagging on each sentence using the python NLTK[2] package. The POS-tagged words $w_1, w_2, ..., w_n$ in each sentence are then mapped to dense embedding vectors $x_1, x_2, ..., x_n$:

A 1D convolutional layer is then applied to the POS representation for each sentence:

$$\mathbf{z}_i = f(\mathbf{W}_z \cdot [\mathbf{x}_i : \mathbf{x}_{i+h_w-1}] + \mathbf{b}_z) \tag{1}$$

where $\mathbf{W}_z$ is a trainable weights matrix, $\mathbf{b}_z$ is a bias vector, and h_w is the size of the convolutional window.

Attention pooling (Dong, Zhang, and Yang 2017) is then applied to learn a sentence representation, shown in Equations 2, 3 and 4.

$$\mathbf{a}_i = tanh(\mathbf{W}_a \cdot \mathbf{z}_i + \mathbf{b}_a) \tag{2}$$

$$u_i = \frac{e^{\mathbf{w}_u \cdot \mathbf{a}_i}}{\sum e^{\mathbf{w}_u \cdot \mathbf{a}_j}} \tag{3}$$

$$\mathbf{s} = \sum u_i \mathbf{z}_i \tag{4}$$

where $\mathbf{W}_a$ and $\mathbf{w}_u$ are a weights matrix and weights vector, $\mathbf{b}_a$ is a bias vector, $\mathbf{a}_i$ and u_i are the attention vector and attention weight for the i-th word, and $\mathbf{s}$ is the final sentence representation.

[2]http://www.nltk.org

Private Layers

In total, there are M tasks, and thus each private layer has M separate copies. The first private layer is a recurrent layer, which is implemented with an LSTM:

$$\mathbf{h}_t^j = \text{LSTM}(\mathbf{s}_{t-1}, \mathbf{s}_t), \quad t = 1, 2, ..., T \tag{5}$$

where the inputs are sentence representations $\{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_T\}$ and $\mathbf{h}_t^j$ is the hidden representation for the j-th task at time-step t.

A private attention-pooling layer is then applied to the hidden representations:

$$\mathbf{e}_t^j = tanh(\mathbf{W}_e^j \cdot \mathbf{h}_t^j + \mathbf{b}_e^j) \tag{6}$$

$$\alpha_t^j = \frac{e^{\mathbf{w}_\alpha^j \cdot \mathbf{e}_t^j}}{\sum e^{\mathbf{w}_\alpha^j \cdot \mathbf{e}_k^j}} \tag{7}$$

$$\mathbf{o}^j = \sum \alpha_t^j \mathbf{h}_t^j \tag{8}$$

where, for the j-th task, $\mathbf{W}_e^j$ and $\mathbf{w}_\alpha^j$ are a weights matrix and weights vector, $\mathbf{b}_e^j$ is a bias vector, $\mathbf{e}_t^j$ and α_t^j are the attention vector and attention weight for the t-th sentence, and $\mathbf{o}^j$ is the final essay representation.

As with PAES (Ridley et al. 2020), a non-prompt-specific features set designed to represent the general essay quality from various perspectives is extracted. We use the same features as PAES in this case, including length-based, readability, text-complexity, text variation, and sentiment-based features. The features set is a vector $\mathbf{f}$, which is concatenated with the representation for each task:

$$\mathbf{c}^j = [\mathbf{o}^j; \mathbf{f}] \tag{9}$$

where $[;]$ denotes the concatenation operation.

To obtain a final representation for each task, we apply a trait-attention mechanism so that each trait can utilize the relevant information from the other trait representations. This mechanism is described by Equations 10 –13.

$$\mathbf{A} = [\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_M] \tag{10}$$

$$v_i^j = \frac{\exp\left(\text{score}(\mathbf{c}^j, A_{-j,i})\right)}{\sum_l \exp\left(\text{score}(\mathbf{c}^j, A_{-j,l})\right)} \tag{11}$$

$$\mathbf{p}^j = \sum v_i^j A_{-j,i} \tag{12}$$

$$\mathbf{g}^j = [\mathbf{c}^j; \mathbf{p}^j] \tag{13}$$

where $\mathbf{A}$ is a concatenation of the representations for each task $\{\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_M\}$, and v_i^j, as calculated in Equation 11, is the attention weight for the i-th trait. We then calculate attention vector $\mathbf{p}^j$ (Equation 12) through a summation of the product of each weight v_i^j and $A_{-j,i}$. The final representation $\mathbf{g}^j$ is a concatenation of $\mathbf{c}^j$ and $\mathbf{p}^j$.

Finally, a linear layer with a sigmoid activation is applied to the representation of each task, as shown in Equation 14.

$$\hat{y}^j = \sigma(\mathbf{w}_y^j \cdot \mathbf{g}^j + b_y^j) \tag{14}$$

where $\hat{y}^j$ is the predicted score for the j-th trait, σ is the sigmoid function, $\mathbf{w}_y^j$ is a weights vector, and b_y^j is a bias.

Set	Num Essays	Traits
1	1783	Content, Organization, Word Choice, Sentence Fluency, Conventions
2	1800	Content, Organization, Word Choice, Sentence Fluency, Conventions
3	1726	Content, Prompt Adherence, Language, Narrativity
4	1772	Content, Prompt Adherence, Language, Narrativity
5	1805	Content, Prompt Adherence, Language, Narrativity
6	1800	Content, Prompt Adherence, Language, Narrativity
7	1569	Content, Organization, Conventions
8	723	Content, Organization, Word Choice, Sentence Fluency, Conventions

Table 1: ASAP and ASAP++ dataset traits

Multi-task Training

Loss We use mean squared error (MSE) as our loss function. Given that there are N essays and M tasks, the loss is calculated as follows:

$$loss(y, \hat{y}) = \frac{1}{NM} \sum_{i=1}^{N} \sum_{j=1}^{M} (\hat{y}_{ij} - y_{ij})^2 \tag{15}$$

Partial Trait Coverage Since the trait set for the i-th training sample is a subset Y_A^i of the entire trait set Y, we need to account for traits without gold scores when calculating the loss. To do this, we introduce a masking function:

$$\forall j \in Y, \quad mask_{ij} = \begin{cases} 1, & \text{if } Y_j \in Y_A^i \\ 0, & \text{otherwise} \end{cases} \tag{16}$$

$$y_i = y_i \otimes mask_i \tag{17}$$

$$\hat{y}_i = \hat{y}_i \otimes mask_i \tag{18}$$

where Y is the set of all possible traits, and $mask_{ij} \in [0, 1]$ for the j-th trait in the i-th essay. Through performing an element-wise multiplication of $mask_i$ with predictions $\hat{y}_i$ and y_i, the trait-wise loss $(\hat{y}_{ij} - y_{ij})^2$ from Equation 15 will be 0 when there is no gold score for y_{ij}.

Experiment Settings

Datasets

In the past, a variety of datasets have been used for English AES (Yannakoudakis, Briscoe, and Medlock 2011; Blanchard et al. 2013; Granger et al. 2011; Stab and Gurevych 2014). However, these are all relatively small-scale datasets. In this work, our experimentation is carried out on the Automated Student Assessment Prize (ASAP)[3] dataset. ASAP is a large-scale dataset that was introduced as part of a Kaggle competition in 2012 and it has since become widely used in prompt-specific AES (Alikaniotis, Yannakoudakis, and Rei 2016; Taghipour and Ng 2016; Dong and Zhang 2016; Dong, Zhang, and Yang 2017; Tay et al. 2018) and cross-prompt AES (Phandi, Chai, and Ng 2015; Cummins, Zhang, and Briscoe 2016; Jin et al. 2018; Ridley et al. 2020) research.

The ASAP dataset contains eight different essay sets, with essays in each set responding to a different prompt. Each essay is awarded a human-rated score for the overall quality of the essay, and the essays for Prompts 7 and 8 are additionally awarded scores for a number of relevant traits according to a scoring rubric.

Each trait has been graded by two raters. We calculate a resolved score through a summation of these two scores, rather than an average, in order to maintain integer values.

Since only Prompts 7 and 8 possess trait scores, we additionally utilize the ASAP++ dataset (Mathias and Bhattacharyya 2018), which builds on top of the original ASAP dataset. The authors of ASAP++ provide scores for various relevant traits for Prompts 1–6 to supplement the original overall scores from ASAP.

The traits for each prompt are presented in Table 1, with trait scores for Prompts 1–6 coming from the supplemented ASAP++ dataset and the trait scores for Prompts 7 and 8 coming from the ASAP dataset. All overall scores are from the original ASAP dataset.

One further piece of pre-processing we perform is to remove traits that appear in only one prompt, because when this prompt is the target prompt, no training samples (i.e. essays from non-target-prompts) will possess scores for the trait. In our case, we remove the unique traits *style* from Prompt 7 and *voice* from Prompt 8. We leave the scoring of novel traits for future work.

Cross Validation

Following research in cross-prompt holistic AES (Jin et al. 2018; Ridley et al. 2020), we perform prompt-wise cross validation, whereby essays for one prompt are used as test data and essays from the remaining prompts are used as training data. This is repeated for each prompt. In each case, the development set comprises essays from the same prompts as the training set.

Evaluation

To evaluate the performance of our model, we use the Quadratic Weighted Kappa (QWK) metric, which has been widely-adopted in both holistic essay scoring and essay trait scoring research (Alikaniotis, Yannakoudakis, and Rei 2016; Cummins, Zhang, and Briscoe 2016; Phandi, Chai, and Ng 2015; Chen and He 2013; Dong and Zhang 2016; Dong, Zhang, and Yang 2017; Jin et al. 2018; Ridley et al. 2020; Mathias and Bhattacharyya 2020; Hussein, Hassan, and Nassef 2020) and is designed to measure the level of agreement between two raters.

[3]The dataset is available at https://www.kaggle.com/c/asap-aes.

	Prompts								
Model	1	2	3	4	5	6	7	8	Avg
Hi att	0.315	0.478	0.317	0.478	0.375	0.357	0.205	0.265	0.349
AES aug	0.330	0.518	0.299	0.477	0.341	0.399	0.162	0.200	0.341
PAES	0.605	0.522	0.575	0.606	**0.634**	0.545	0.356	0.447	0.536
CTS no att	0.619	0.539	0.585	0.616	0.616	0.544	0.363	0.461	0.543
CTS	**0.623**	**0.540**	**0.592**	**0.623**	0.613	**0.548**	**0.384**	**0.504**	**0.553**

Table 2: Average QWK scores across all traits for each prompt on ASAP/ASAP++ dataset

	Traits									
Model	*Overall*	*Content*	*Org*	*WC*	*SF*	*Conv*	*PA*	*Lang*	*Nar*	Avg
Hi att	0.453	0.348	0.243	0.416	0.428	0.244	0.309	0.293	0.379	0.346
AES aug	0.402	0.342	0.256	0.402	0.432	0.239	0.331	0.313	0.377	0.344
PAES	0.657	0.539	0.414	0.531	0.536	0.357	**0.570**	0.531	0.605	0.527
CTS no att	0.659	0.541	0.424	**0.558**	0.544	0.387	0.561	**0.539**	0.605	0.535
CTS	**0.670**	**0.555**	**0.458**	0.557	**0.545**	**0.412**	0.565	0.536	**0.608**	**0.545**

Table 3: Average QWK scores across all prompts for each trait on ASAP/ASAP++ dataset: Due to space limitations, some trait names have been simplified—*Org* refers to organization, *WC* to word choice, *SF* to sentence fluency, *Conv* to conventions, *PA* to prompt adherence, *Lang* to language and *Nar* to narrativity.

Baselines

We compare our approach with four strong baselines, each of which is described below:

- *Hi att*: This model is a leading prompt-specific holistic scoring model (Dong, Zhang, and Yang 2017) and is the best performing model tested by Mathias and Bhattacharyya (2020) for prompt-specific trait scoring. The model captures the hierarchical structure of the essays and utilises a convolutional layer with attention pooling to capture sentence-level representations followed by a recurrent layer and another attention pooling layer to capture the essay-level representation. As with Mathias and Bhattacharyya (2020), we use a single-task architecture, training the model on each trait individually.
- *AES aug*: This model (Hussein, Hassan, and Nassef 2020) is a trait-scoring model, which converts the holistic scoring model from Taghipour and Ng (2016) into a multi-task approach through adding a linear layer for each trait on top of the original encoder from Taghipour and Ng (2016).
- *PAES*: This is a leading cross-prompt holistic scoring model (Ridley et al. 2020) and is the model on which our approach is based. The architecture is depicted in the left-hand-side image of Figure 2. As with *Hi att*, training is performed on each trait individually.
- *CTS no att*: This model uses the multi-task architecture of our *CTS* model, utilizing the shared- and private-layer architecture as depicted in the right-hand-side image of Figure 2, but without the use of the trait-attention mechanism.

Implementation Details

We first convert all the words into lower case and tokenize them using NLTK tokenizer. For the *Hi att* and *AES aug* models, we use 50-dimension pre-trained GloVe word embeddings (Pennington, Socher, and Manning 2014). For *PAES*, *CTS no att* and *CTS*, we learn 50-dimension POS embeddings.

Optimization for all models is carried out with the RMSprop algorithm (Dauphin, de Vries, and Bengio 2015) with the learning rate set to 0.001.

We train all models for a total of 50 epochs. For the single-task approaches, the best epoch is the epoch for which the development dataset has the highest QWK score for the target trait. We then report the QWK score for the test set on the same epoch. For multi-task approaches, we consider all traits to be equally important so the best epoch is determined by taking the highest mean QWK score over all traits.

All models are implemented with Tensorflow[4] in Python. We run each model five times on an NVIDIA[5] GeForce GTX 1080 graphics card; we report the mean scores across the five runs.

Results and Analyses

We report the results for our experiments across two dimensions. In Table 2, we display the average score across all traits for each prompt, and in Table 3 we display the average score across all prompts for each trait.

From looking at both Tables 2 and 3, we can see that the two prompt-specific trait-scoring models (*Hi att* and *AES aug*) both perform poorly across both dimensions. This is due to the fact that these models are not designed for cross-prompt scoring and thus they both overfit considerably.

When we compare the three cross-prompt models, we can see that *CTS* outscores both *PAES* and *CTS no att* on all but Prompt 5 (Table 2) and also that the multi-task approach *CTS no att* performs better than the single-task approach *PAES* on most prompts. This is due to the fact that *PAES* is unable to

[4]https://www.tensorflow.org/

[5]https://www.nvidia.com/

	Traits						
Model	*Overall*	*Content*	*Organisation*	*Word Choice*	*Sent Fluency*	*Conventions*	Avg
PAES	0.593	**0.576**	0.496	0.480	0.534	0.453	0.522
CTS no att	0.578	0.558	0.498	**0.544**	**0.567**	**0.488**	0.539
CTS	**0.617**	0.518	**0.514**	0.534	**0.567**	**0.488**	**0.540**

Table 4: Average QWK scores for Prompt 2 for each trait on ASAP/ASAP++ dataset

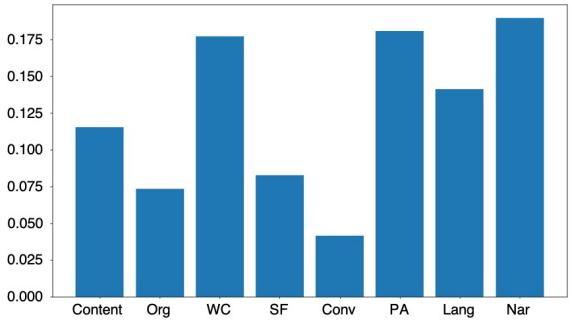

Figure 3: Attention weights for all traits when predicting *overall* score for Prompt 3

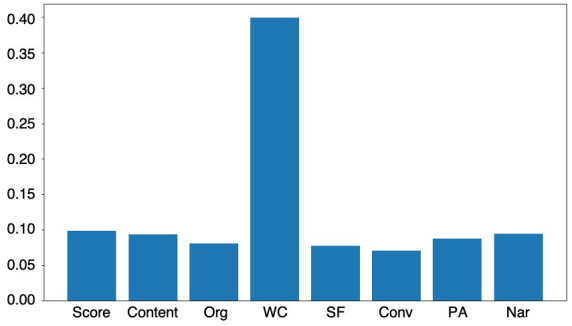

Figure 4: Attention weights for all traits when predicting the *language* score for Prompt 3

utilize the full training dataset, as it can only train on samples for which there is a gold score for the target trait. This is not the case for either *CTS no att* or *CTS*, which, due to their multi-task architectures, are able to utilize all samples in the training set.

Effect of Limited Number of Samples for Target Traits

Something that seems to have an impact on the performance of the *PAES* model is the number of training data available. Displayed in Table 4 is the performance for each of the traits of Prompt 2 for *PAES*, *CTS no att* and *CTS*. In this table, there are two traits, *Word Choice* and *Sentence Fluency*, which appear in only two other prompts. As a result, the single-task-based approach is only able to train on 2129 essays of the 9499 essays in the training set when scoring these particular traits. This leads to the performance for these two traits being significantly reduced when compared with the other two models.

Effect of Trait Attention

In order to gain some insight into the workings of our trait-attention mechanism, we visualize the attention weights. In Figure 3, the attention weights are displayed for all traits when predicting the *overall* score for Prompt 3. This is a holistic score, which should consider the quality of the essay from a variety of different aspects. From the image, we can see that the attention is distributed fairly evenly throughout the traits, without any one trait having a significantly higher score than the others.

In contrast, Figure 4 displays the trait attention weights when predicting the *language* score for Prompt 3. This is a specific and more focused trait than the *overall* score. Here, we can see that the attention weights for most traits are fairly low, except for the trait *word choice*, which is closely related to *language*, and has a considerably higher weight than the others.

Conclusion and Future Work

To address the needs of real-world AES systems, namely that there is a need to be able to score essays to novel prompts and that feedback across different trait dimensions is desirable for users, we introduce a new AES task named Automated Cross-prompt Scoring of Essay Traits.

In addition, we introduce a new method called Cross-prompt Trait Scorer (CTS) that utilizes a multi-task architecture with shared and private layers along with a trait-attention mechanism to address the issues of limited training data for certain traits in the cross-prompt setting and inter-trait relationships.

Given that in this paper we choose not to address the issue of scoring novel traits, we believe that our work could further be improved by exploring methods of scoring unseen traits. We leave this task for future work in the domain of Cross-prompt Scoring of Essay Traits.

Acknowledgements

We thank all anonymous reviewers who provided valuable feedback. Our research was supported by the National Key R&D Program of China (No. 2018YFB1005102) and the NSFC (No. 61976114, 61936012).

References

Alikaniotis, D.; Yannakoudakis, H.; and Rei, M. 2016. Automatic Text Scoring Using Neural Networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 715–725. Association for Computational Linguistics.

Attali, Y.; Bridgeman, B.; and Trapani, C. 2010. Performance of a generic approach in automated essay scoring. *The Journal of Technology, Learning and Assessment* 10(3).

Blanchard, D.; Tetreault, J.; Higgins, D.; Cahill, A.; and Chodorow, M. 2013. TOEFL11: A CORPUS OF NON-NATIVE ENGLISH. *ETS Research Report Series* 2013(2): i–15.

Burstein, J.; Kukich, K.; Wolff, S.; Lu, C.; Chodorow, M.; Braden-Harder, L.; and Harris, M. D. 1998. Automated Scoring Using A Hybrid Feature Identification Technique. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, 206–210. Association for Computational Linguistics.

Chen, H.; and He, B. 2013. Automated Essay Scoring by Maximizing Human-Machine Agreement. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 1741–1752. Association for Computational Linguistics.

Cummins, R.; Zhang, M.; and Briscoe, T. 2016. Constrained Multi-Task Learning for Automated Essay Scoring. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 789–799. Association for Computational Linguistics.

Dauphin, Y.; de Vries, H.; and Bengio, Y. 2015. Equilibrated adaptive learning rates for non-convex optimization. In *Advances in Neural Information Processing Systems*, volume 28, 1504–1512. Curran Associates, Inc.

Dong, F.; and Zhang, Y. 2016. Automatic Features for Essay Scoring – An Empirical Study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1072–1077. Association for Computational Linguistics.

Dong, F.; Zhang, Y.; and Yang, J. 2017. Attention-based Recurrent Convolutional Neural Network for Automatic Essay Scoring. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, 153–162. Association for Computational Linguistics.

Farra, N.; Somasundaran, S.; and Burstein, J. 2015. Scoring Persuasive Essays Using Opinions and their Targets. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, 64–74. Association for Computational Linguistics.

Granger, S.; Dagneaux, E.; Meunier, F.; ; and Magali Paquot (Eds.). Louvain-La-Neuve, France: Presses Universitaires de Louvain, . P. . 2011. INTERNATIONAL CORPUS OF LEARNER ENGLISH: VERSION 2. *Studies in Second Language Acquisition* 33(1): 140–142.

Hochreiter, S.; and Schmidhuber, J. 1997. Long short-term memory. *Neural Computation* 9(8): 1735–1780.

Hussein, M. A.; Hassan, H. A.; and Nassef, M. 2020. A Trait-based Deep Learning Automated Essay Scoring System with Adaptive Feedback. *International Journal of Advanced Computer Science and Applications* 11(5).

Jin, C.; He, B.; Hui, K.; and Sun, L. 2018. TDNN: A Two-stage Deep Neural Network for Prompt-independent Automated Essay Scoring. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1088–1097. Association for Computational Linguistics.

Ke, Z.; Carlile, W.; Gurrapadi, N.; and Ng, V. 2018. Learning to Give Feedback: Modeling Attributes Affecting Argument Persuasiveness in Student Essays. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, 4130–4136. International Joint Conferences on Artificial Intelligence Organization.

Mathias, S.; and Bhattacharyya, P. 2018. ASAP++: Enriching the ASAP Automated Essay Grading Dataset with Essay Attribute Scores. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. European Language Resources Association (ELRA).

Mathias, S.; and Bhattacharyya, P. 2020. Can Neural Networks Automatically Score Essay Traits? In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, 85–91. Association for Computational Linguistics.

Miltsakaki, E.; and Kukich, K. 2004. Evaluation of text coherence for electronic essay scoring systems. *Natural Language Engineering* 10(1): 25–55.

Pennington, J.; Socher, R.; and Manning, C. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. Association for Computational Linguistics.

Persing, I.; Davis, A.; and Ng, V. 2010. Modeling Organization in Student Essays. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, 229–239. Association for Computational Linguistics.

Persing, I.; and Ng, V. 2013. Modeling Thesis Clarity in Student Essays. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 260–269. Association for Computational Linguistics.

Persing, I.; and Ng, V. 2014. Modeling Prompt Adherence in Student Essays. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1534–1543. Association for Computational Linguistics.

Phandi, P.; Chai, K. M. A.; and Ng, H. T. 2015. Flexible Domain Adaptation for Automated Essay Scoring Using Correlated Linear Regression. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 431–439. Association for Computational Linguistics.

Ridley, R.; He, L.; Dai, X.-Y.; Huang, S.; and Chen, J. 2020. Prompt Agnostic Essay Scorer: A Domain Generalization Approach to Cross-prompt Automated Essay Scoring. *ArXiv* abs/2008.01441.

Rudner, L. M.; and Liang, T. 2002. Automated Essay Scoring Using Bayes' Theorem. *The Journal of Technology, Learning and Assessment* 1(2).

Sanh, V.; Wolf, T.; and Ruder, S. 2019. A Hierarchical Multi-Task Approach for Learning Embeddings from Semantic Tasks. *Proceedings of the AAAI Conference on Artificial Intelligence* 33(01): 6949–6956.

Stab, C.; and Gurevych, I. 2014. Annotating Argument Components and Relations in Persuasive Essays. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, 1501–1510. Dublin City University and Association for Computational Linguistics.

Taghipour, K.; and Ng, H. T. 2016. A Neural Approach to Automated Essay Scoring. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1882–1891. Association for Computational Linguistics.

Tay, Y.; Phan, M.; Tuan, L. A.; and Hui, S. C. 2018. SkipFlow: Incorporating Neural Coherence Features for End-to-End Automatic Text Scoring. *Proceedings of the AAAI Conference on Artificial Intelligence* 5948–5955.

Wachsmuth, H.; Al-Khatib, K.; and Stein, B. 2016. Using Argument Mining to Assess the Argumentation Quality of Essays. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 1680–1691. The COLING 2016 Organizing Committee.

Yannakoudakis, H.; Briscoe, T.; and Medlock, B. 2011. A New Dataset and Method for Automatically Grading ESOL Texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 180–189. Association for Computational Linguistics.