# PLAES: Prompt-generalized and Level-aware Learning Framework for Cross-prompt Automated Essay Scoring

**Xia Li and Yuan Chen***
School of Information Science and Technology,
Guangdong University of Foreign Studies, Guangzhou, China
{xiali, yuanchen}@gdufs.edu.cn

**Abstract**

Current cross-prompt automatic essay scoring (AES) systems are primarily concerned with obtaining shared knowledge specific to the target prompt by using the source and target prompt essays. However, it may not be feasible in practical situations because the target prompt essays may not be available as training data. When constructing a model solely from source prompt essays, its capacity to generalize to the target prompt may be hindered by significant discrepancies among different prompt essays. In this study, a novel learning framework for cross-prompt AES is proposed in order to capture more general knowledge across prompts and improve the model's capacity to distinguish between writing levels. To acquire generic knowledge across different prompts, a primary model is trained via meta learning with all source prompt essays. To improve the model's ability to differentiate writing levels, we present a level-aware learning strategy consisting of a general scorer and three level scorers for low-, middle-, and high-level essays. Then, we introduce a contrastive learning strategy to bring the essay representation of the general scorer closer to its corresponding level representation and far away from the other two levels, thereby improving the system's ability to differentiate writing levels as well as boosting scoring performance. Experimental results on public datasets illustrate the efficacy of our method.

**Keywords:** Automated Essay Scoring, Cross-prompt AES, Meta Learning, Contrastive Learning

## 1. Introduction

Writing skills is important for students and language learners. Automated Essay Scoring (AES) aims to judge the quality of a student's writing automatically. In comparison to the human grading process, a comprehensive AES system can not only reduce the workload of human raters, but also improve grading consistency and scoring fairness (Hearst, 2000; Weigle, 2002; Uto et al., 2020).

Previous AES research focused primarily on developing a model for assessing the quality of essays written in response to a specific prompt[1] (Prompt-specific AES). Earlier works employed rich elaborate hand-crafted features to create efficient scoring models (Weigle, 2002; Attali and Burstein, 2006; Persing and Ng, 2013; Sultan et al., 2016). With the rise of deep learning, more studies (Taghipour and Ng, 2016; Dong et al., 2017; Tay et al., 2018; Hussein et al., 2020; Uto et al., 2020; Liao et al., 2021; Wang et al., 2022; Xie et al., 2022; He et al., 2022) have investigated the application of neural networks for prompt-specific AES with promising results.

In real-world application situations, we may only receive labeled essays from source prompts but are unable to obtain them from target prompts or can only obtain a small part of them. In response to this circumstance, cross-prompt AES systems have been proposed in recent years. These methods can be broadly classified into three categories. The first class of approaches uses source prompt essays and a small sample of labeled target prompt essays to train the model and learn about the knowledge specific to the target prompt (Phandi et al., 2015; Cummins et al., 2016; Song et al., 2020). In contrast, the second class of methods does not include any labeled target prompt essays. In these studies, such as the work of Jin et al. (2018); Li et al. (2020); Cao et al. (2020) and Chen and Li (2023), unlabeled target prompt essays are employed to obtain transferable knowledge. The core idea behind these methods is to use both labeled source prompt essays and unlabeled target prompt essays to build the model and capture more shared knowledge across the source and target prompt essays.

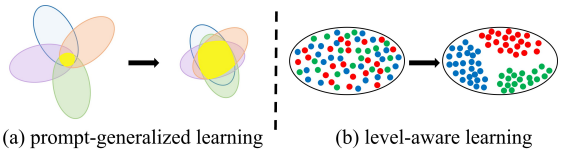


Figure 1: A summary of our motivations. In (a), circles with different colors denote different source prompts, and the yellow area denotes the general knowledge across these prompts. In (b), red, green and blue points represent essays of low-, middle- and high-levels in target prompt.

Although the two types of approaches described above are effective, they still need to see the tar-

* Corresponding author.
[1]The prompt refers to the writing theme of essays.



*LREC-COLING 2024*, pages 12775–12786

get prompt essays, which may not be feasible in practice. To this end, the third type of work uses essays only from source prompts to train the model without seeing any target prompt essays. This kind of method has the benefit of being able to grade essays on different new prompts with only one training session. Only a limited number of studies (Ridley et al., 2020, 2021; Jiang et al., 2023) have been carried out based on this type of setting. Among the existing work, Ridley et al. (2020, 2021) propose to utilize the non prompt-specific handcrafted features (e.g., sentence length and part-of-speech tag counts) to capture the essay quality from different aspects to improve generalization to new prompts. However, we believe that the huge discrepancies between different source prompts may result in low generalization to new prompts, which may not be fully captured by the handcrafted linguistic features. Jiang et al. (2023) propose a prompt-aware neural AES model to extract prompt-invariant and prompt-specific features. We believe that extracting prompt-aware information is not enough, and other information related to the scoring task, such as writing levels, should be considered.

The majority of previous cross-prompt AES systems have so far been concerned with grading essays according to their scores using regression-based constraint loss. As stated in previous work (Jin et al., 2018), intuitively, essays with good quality tend to have a higher score range, while those with poor quality tend to fall into the lower score range. Therefore, we believe that the writing levels (e.g., low, middle, and high) can be considered as a general and consistent evaluation of the quality of essays on different prompts, which can be a complement to the quality evaluation of the essays and further improve the model's generalization to new prompts.

Our task of cross-prompt AES using essays only from source prompts as training data exhibits two challenges. First, how to obtain more common and general neural features across all prompts to better express the quality of essays for new prompts due to the huge difference between different source prompts. Second, how to employ the writing level as a complement constraint to learn more features to differentiate the essay's quality and improve the model's generalization to new prompts.

In this paper, we design a **P**rompt-generalized and **L**evel-aware learning framework for cross-prompt **A**utomated **E**ssay **S**coring (PLAES). For the first challenge, as Figure 1 (a), we design a prompt-generalized learning strategy based on meta learning to capture more general knowledge across different source prompts. In this way, we are able to obtain a primary representation for all essays from all source prompts. To cope with the second challenge, as Figure 1 (b), we propose a level-aware learning strategy to improve the model's capacity to differentiate essay quality under the constraints of writing levels. Specifically, we design a general scoring model and three level scoring models for low-, middle-, and high-level essays, where the level scoring models are used to learn level-specific scoring knowledge. Then, we construct a contrastive learning strategy to bring the essay representation of the general scoring model closer to its corresponding level representation and further away from the representations of the other two levels, thereby enhancing the model's ability to distinguish writing levels and boosting the scoring performance.

The summarization of our contributions is as follows:

(1) To the best of our knowledge, this is the first attempt to explore the use of writing level as a supplement constraint for regression-based constraints for cross-prompt AES in order to better differentiate essay quality and improve the model's generalization to new prompts.

(2) We present a prompt-generalized scoring model for cross-prompt AES that uses only source prompt essays as training data and ensures essay quality with more general neural features.

(3) Experimental results on the public datasets show that our proposed method outperforms all baseline models.

## 2. Related Work

### 2.1. Automated Essay Scoring

Most of the AES studies focus on prompt-specific settings, which train and test models on the same prompt. Some researchers (Rudner and Liang, 2002; Attali and Burstein, 2006; Mohler and Mihalcea, 2009; Persing and Ng, 2013; Sultan et al., 2016; Salim et al., 2019) score essays by extracting relevant features and analyzing the quality of the essays contained in the features with machine learning algorithms. Most recent work (Dong and Zhang, 2016; Taghipour and Ng, 2016; Dong et al., 2017; Tay et al., 2018; Hussein et al., 2020; Uto et al., 2020; Liao et al., 2021; He et al., 2022; Shibata and Uto, 2022; Wang et al., 2022; Xie et al., 2022; Wang et al., 2023; Ding et al., 2023) use deep learning models to extract richer semantic features from essays and achieve better results.

Another setting is cross-prompt AES, in which a model is trained on a labeled source prompt and tested on an unlabeled target prompt. The first class of approaches (Phandi et al., 2015; Cummins et al., 2016; Song et al., 2020) uses source prompt essays and a small sample of labeled target prompt essays to train the model and learn about the knowledge specific to the target prompt. The

second class of methods (Jin et al., 2018; Li et al., 2020; Cao et al., 2020; Chen and Li, 2023) does not include any labeled target prompt essays but uses labeled source prompt essays and unlabeled target prompt essays as training data. Different from the above two types of methods, the third type of work (Ridley et al., 2020, 2021; Do et al., 2023; Jiang et al., 2023) uses essays only from source prompts to train the model without any target prompt essays.

### 2.2. Meta Learning

Meta learning is a training strategy that enables the acquisition of generic knowledge from diverse sources and adaptation to novel domains. Existing meta learning methods can be broadly classified into two categories: gradient-based approaches (Finn et al., 2017; Mi et al., 2019; Yan et al., 2020; Yao et al., 2021; Nan et al., 2022; Li et al., 2023) and metric-based methods (Yao et al., 2021). The former aim to transfer knowledge across tasks during meta training, while the latter concentrate on developing a distance metric to assess the similarity between data pairs. In this paper, we follow the MAML (Finn et al., 2017) to train prompt-generalized learning in Step I.

### 2.3. Contrastive Learning

Due to the ability to learn effective representations, contrastive learning has gained enormous popularity recently, such as SimCLR (Chen et al., 2020). The fundamental step in contrastive learning lies in constructing positive samples. Data augmentation is a widely used positive example construction method in Natural Language Processing. Some work obtains the positive examples by partial modification of the original text (Wang et al., 2021; Han et al., 2022; Liang et al., 2022; Wu et al., 2022), while other researchers obtain the desired positive examples by perturbation of the representation of the text (Gao et al., 2021; Jiang et al., 2022; Zhang et al., 2022; Chen and Li, 2023).

## 3. Our Approach

Our approach consists of three key components: 1) Scoring model is used to obtain the encoding representation of essays and predict a set of attribute scores. 2) Prompt-generalized learning module attempts to obtain more general representations for the general scoring model and three level scoring models. 3) Level-aware learning module aims to improve the model's capacity to differentiate essay quality based on writing level constraints. The overview of our approach is shown in Figure 2.

### 3.1. Task Definition

Given source prompt data $P = \{P_i\}_{i=1}^{N}$, where $N$ is the number of source prompts. Each prompt consists of a number of essays, each with an essay text x and a set of attribute scores $Y = \{y_a\}_{a=1}^{A}$, where A is the number of attributes and y0 represents the total score. The model accepts x as input and uses Y as the optimization label for the scoring task. The task of our approach is to train a model with $P$ and evaluate it on an unseen target prompt. The complete procedure is shown in Algorithm 1.

### 3.2. Scoring Model

The scoring model consists of essay encoder and essay scorer. The encoder is used to obtain the essay representations, while the scorer is used to predict the scores of the essay's multiple attributes.

**Essay Encoder** In this paper, we employ a hierarchical structure (Dong et al., 2017) as encoder. The encoder captures the sentence representation of all words in each sentence, as well as the essay representation of all sentences. Assuming that each sentence consists of $n$ words, the sentence representation $s$ can be extracted by CNN (Kim, 2014) and attention pooling (Sutskever et al., 2014) from a sequence of words $\{w_1, w_2, ..., w_m\}$. Following previous work, we use the embeddings of Part-of-Speech (POS)[2] to represent the essay text. For convenience, we use $w_i$ to denote POS embeddings. The sentence representation $s$ can be obtained as follows:

$$c_i = \text{CNN}([w_i : w_{i+k-1}]), i = 1, 2, ..., m, \quad (1)$$

$$s = \text{pooling}([c_1 : c_m]), \quad (2)$$

where $k$ is the kernel size of CNN.

Then, the essay representation $h$ can be extracted by the LSTM (Hochreiter and Schmidhuber, 1997) and another attention pooling from all sentence representations $\{s_1, s_2, ..., s_n\}$. The equations are as follows:

$$r_i = \text{LSTM}(s_{i-1}, s_i), \ i = 1, .., n, \quad (3)$$

$$h = \text{pooling}([r_1 : r_n]), \quad (4)$$

where $r_t$ represents the output of LSTM at the $t$-th time step, and $h$ is the final essay representation.

**Essay Scorer** As we need to predict a set of attribute scores for an essay, we first setup a specific relu-dense layer for each attribute and take $h$ as input to obtain multiple attribute representations $h_a \in \{h_1, h_2, ..., h_A\}$, where $A$ is the number of attributes. The final representation is obtained by

[2]We use NLTK (http://www.nltk.org) toolkit.

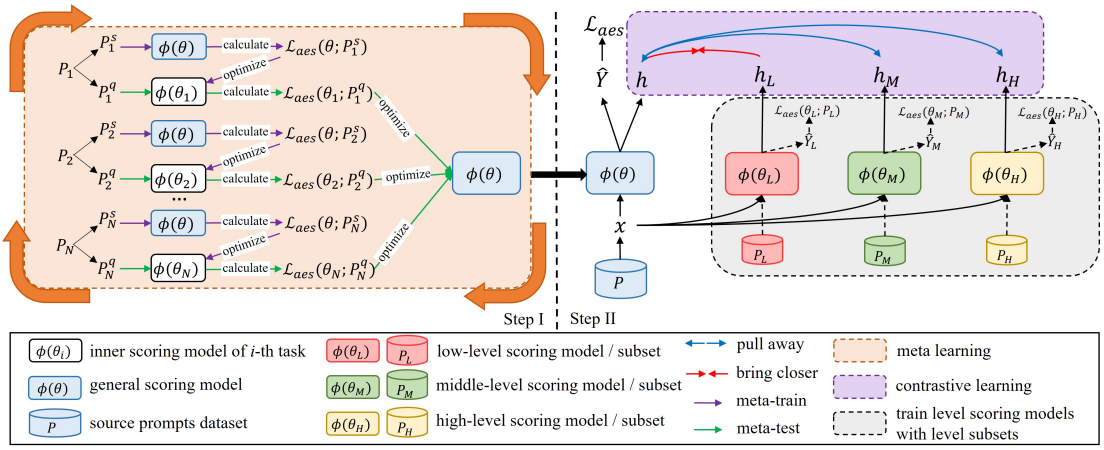


Figure 2: Overview of our approach. The Step I is to train prompt-generalized learning via meta learning, and the Step II is to train level-aware learning via contrastive learning and fine-tune the scoring task simultaneously. $x$ in Step II is assumed to be a low-level essay.

concatenating the handcrafted features $\mathbf{f}$ proposed by Ridley et al. (2021) with the attribute representation $h_a$. Then, another specific sigmoid-dense layer is utilized to predict an attribute-specific score $\hat{y}_a$. Finally, we can get all predicted scores for all attributes $\hat{Y} = \{\hat{y}_a\}_{a=1}^{A}$. The corresponding equations are as follows:

$$h_a = relu(W_h \cdot h + b_h), \tag{5}$$

$$\hat{y}_a = sigmoid(W_y \cdot [h_a; \mathbf{f}] + b_y), \tag{6}$$

where $W_h$ and $W_y$ are the trainable weight matrices, $b_h$ and $b_y$ are the bias vectors, and $[;]$ denotes concatenate operation.

Finally, we use mean squared error as scoring loss function.

$$\mathcal{L}_{aes} = \frac{1}{A}\sum_{a=1}^{A}(\hat{y}_a - y_a)^2. \tag{7}$$

It is important to note that not every essay has the same set of attributes. In order to train the model on essays with different attributes, we use a mask mechanism proposed by Ridley et al. (2021) to mask the missing attributes in the essay.

$$mask_a = \begin{cases} 1, \ if \ y_a \in Y \\ 0, \ otherwise \end{cases}. \tag{8}$$

**Algorithm 1:** Procedure of PLAES

**Input:** source prompts $P = \{P_i\}_{i=1}^{N}$
**Output:** general scoring model $\phi(\theta)$

Randomly initialize $\theta$;
**for** *prompt-generalized iteration* **do**
  Sample a batch of training tasks $T = \{T_i\}_{i=1}^{N}$;
  **for** $T_i \in \{T_1, T_2, ..., T_N\}$ **do**
    Sample $P_i^s, P_i^q$ from $P_i$;
    Calculate $\mathcal{L}_{aes}(\theta; P_i^s)$;
    Update $\phi(\theta_i)$ by Eq. (9);
    Calculate $\mathcal{L}_{aes}(\theta_i; P_i^q)$;
  Update $\theta$ by Eq. (10);
Initialize level scoring models:
$\theta_L, \theta_M, \theta_H = \theta$;
**for** *level-aware iteration* **do**
  Train level scoring models by Eq. (11);
  Sample a batch of data $x$;
  **for** $x \in P$ **do**
    $h = \phi(\theta; x)$;
    $h_o = \phi(\theta_o; x), o \in L, M, H$;
    Calculate $\mathcal{L}_{aes}$ by Eq. (7);
    Calculate $\mathcal{L}_{LA}$ by Eq. (12);
    Update $\theta$ by optimizing $\mathcal{L}_{aes}$ and $\mathcal{L}_{LA}$;

### 3.3. Prompt-generalized Learning

To obtain more general knowledge across source prompts, we apply meta learning to train the general scoring model. Specifically, we take each prompt $P_i$ in $N$ source prompts $P = \{P_i\}_{i=1}^{N}$ as a separate task for meta learning, and then we update the general scoring model by incorporating prompt-specific scoring knowledge from all prompts.

For each iteration of meta learning, a batch of training tasks $T = \{T_i\}_{i=1}^{N}$ are sampled from source prompts. In each task $T_i$, we sample a support set $P_i^s$ and a query set $P_i^q$ from $P_i \in P$. Assuming that the general scoring model parameters are denoted as $\theta$ and the task-specific parameters of the inner scoring model are denoted as $\theta_i \in \{\theta_1, \theta_2, ..., \theta_N\}$. For each task $T_i$, $\theta_i$ can be updated by training models with support set $P_i^s$,

| Prompt ID | No. of Essays | Avg. Length | Essay Type | Grade Level | Attributes | Score Range | |
|---|---|---|---|---|---|---|---|
| | | | | | | Overall | Attribute |
| P1 | 1,783 | 350 | Argumentative | 8 | Cont, Org, WC, SF, Conv | 2 - 12 | 1 - 6 |
| P2 | 1,800 | 350 | Argumentative | 10 | Cont, Org, WC, SF, Conv | 0 - 6 | 1 - 6 |
| P3 | 1,726 | 150 | Source-Dependent | 10 | Cont, PA, Lan, Nar | 0 - 3 | 0 - 3 |
| P4 | 1,772 | 150 | Source-Dependent | 10 | Cont, PA, Lan, Nar | 0 - 3 | 0 - 3 |
| P5 | 1,805 | 150 | Source-Dependent | 8 | Cont, PA, Lan, Nar | 0 - 4 | 0 - 4 |
| P6 | 1,800 | 150 | Source-Dependent | 10 | Cont, PA, Lan, Nar | 0 - 4 | 0 - 4 |
| P7 | 1,569 | 300 | Narrative | 7 | Cont, Org, Conv | 0 - 30 | 0 - 6 |
| P8 | 723 | 650 | Narrative | 10 | Cont, Org, WC, SF, Conv | 0 - 60 | 2 - 12 |

Table 1: Statistics of ASAP++ Dataset. Cont denotes *Content*, Org is *Organization*, WC is *Word Choice*, SF is *Sentence* Fluency, Conv is *Conventions*, PA is *Prompt Adherence*, Lan is *Language*, Nar is *Narrativity*.

which can be called meta-train process.

$$\theta_i = \theta - \alpha \bigtriangledown_\theta \mathcal{L}_{aes}(\theta; P_i^s), \tag{9}$$

where $\alpha$ is the learning rate of meta-train process, and $\mathcal{L}_{aes}(\theta; P_i^s)$ is the scoring loss calculated by using scoring model with parameters $\theta$ and $P_i^s$.

Then, we calculate meta-test loss $\mathcal{L}_{aes}(\theta_i; P_i^q)$ with $\theta_i$ and query set $P_i^q$, which is used as training loss of meta learning on the current task.

After loop all tasks, the parameters $\theta$ of the general scoring model can be updated as follows:

$$\theta = \theta - \beta \bigtriangledown_\theta \sum_i^N \mathcal{L}_{aes}(\theta_i; P_i^q), \tag{10}$$

where $\beta$ is the learning rate of meta-test process and $N$ is the number of source prompts.

### 3.4. Level-aware Learning

To improve model's capacity to differentiate essays quality within the constraints of writing level, we design a level-aware contrastive learning strategy to bring the essay representation of general scoring model closer to the representation of its corresponding level and further away from the representations of the other two levels. The details are as follows.

Firstly, inspired by Jin et al. (2018), we normalize essay's overall scores on [0, 1] scale. Essays with [0, 0.4] represent low-level essays, (0.4, 0.8) represent middle-level essays, and [0.8, 1] represent high-level essays. These level subsets can be denoted as $P_L, P_M, P_H$. Then, for each level, we initialize a level scoring model with the same parameters as the general scoring model, denoted as $\theta_o = \theta, o \in \{L, M, H\}$. At the beginning of each iteration, the level scoring models are updated via training scoring tasks with level subsets in order to acquire the scoring knowledge from each level.

$$\theta_o = \theta_o - \gamma \bigtriangledown_{\theta_o} \mathcal{L}_{aes}(\theta_o; P_o), \tag{11}$$

where $\gamma$ is the learning rate of training level models.

After that, we sample a batch of essays from all source prompts $P$ to train the level-aware learning and fine-tune scoring tasks. For convenience (as shown in Figure 2), we introduce the training process with an essay $x \in \{x_L, x_M, x_H\}$. We input $x$ into three level scoring models and the general scoring model to get level representations $h_L, h_M, h_H$ and general representation $h$, denoted as $h_o = \phi(\theta_o; x), o \in \{L, M, H\}$ and $h = \phi(\theta; x)$. For low-level essay $x_L$, its corresponding positive example is $h_L$, while the negative examples are $h_M$ and $h_H$. The level-aware contrastive learning loss for $x_L$ is constructed as:

$$\mathcal{L}_{LA}^L = -log \frac{f(h, h_L)}{\sum_{o \in \{L,M,H\}} f(h, h_o)}, \tag{12}$$

where $f(a, b) = exp(cos(a, b)/\tau)$, $cos(\cdot)$ is the cosine similarity function and $\tau$ is the temperature. Finally, the total loss function of $x_L$ in Step II is:

$$\mathcal{L}^L = \mathcal{L}_{aes}(\theta; x_L) + \lambda \mathcal{L}_{LA}^L, \tag{13}$$

where $\lambda$ is the weighted hyper-parameter. Similarly, the losses for middle-level essay $x_M$ and high-level essay $x_H$ can be calculated in the same way:

$$\mathcal{L}^M = \mathcal{L}_{aes}(\theta; x_M) + \lambda \mathcal{L}_{LA}^M, \tag{14}$$

$$\mathcal{L}^H = \mathcal{L}_{aes}(\theta; x_H) + \lambda \mathcal{L}_{LA}^H. \tag{15}$$

The final batch loss is the mean of all essay losses in the batch.

## 4. Experiments

### 4.1. Dataset and Evaluation Metric

We conduct experiments on the ASAP++ (Mathias and Bhattacharyya, 2018) dataset, which is an eight-prompt, multi-attribute essay scoring dataset derived from the ASAP[3] dataset. Each essay receives an overall score as well as multiple attribute scores, and the statistics of dataset are provided in Table 1. It should be noted that there are unique attributes *Style* in P7 and *Voice* in P8. Following Ridley et al. (2021), we removed them in this paper.

For each target prompt, the remaining seven prompts serve as source prompts. For example, if

[3]https://www.kaggle.com/c/asap-aes

| Model | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | 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 |
| PMAES | **0.656** | 0.553 | 0.598 | 0.606 | 0.626 | 0.572 | 0.386 | 0.530 | 0.566 |
| PLAES (ours) | 0.648 | **0.563** | **0.604** | **0.623** | **0.634** | **0.593** | **0.403** | **0.533** | **0.575** |

Table 2: Average QWK of all attributes on each prompt.

| Model | Overall | Cont | Org | WC | SF | Conv | PA | Lan | 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 |
| PMAES | 0.671 | 0.567 | 0.481 | **0.584** | **0.582** | 0.421 | 0.584 | 0.545 | 0.614 | 0.561 |
| PLAES (ours) | **0.673** | **0.574** | **0.491** | 0.579 | 0.580 | **0.447** | **0.601** | **0.554** | **0.631** | **0.570** |

Table 3: Average QWK for each attribute over all prompts.

target prompt is P1 and source prompts are P2 to P8, then the training and validation sets are from P2 to P8, and the test set is P1.

The evaluation metric is Quadratic Weighted Kappa (QWK). It is a commonly used evaluation metric in AES and use to assess the consistency between actual and predicted scores.

### 4.2. Experiment Setting

Our training data are only from source prompt. In Step I of the training procedure, we sample the meta learning tasks from source prompt for prompt-generalized learning. In Step II, we use a batch of samples from source prompt to train level-aware learning and fine-tune the scoring task.

All scoring models use 50-dim POS embeddings as input. The kernel size is 5, the number of filters is 100 in CNN, and the number of units is 100 in LSTM. In all steps, the optimizers are RMSprop (Dauphin et al., 2015). The handcrafted features are 86-dimensional features from Ridley et al. (2020), including features of Length-based, Readability, Text Complexity, Text Variation and Sentiment.

For prompt-generalized learning, the number of meta learning iterations is 1000, and the batch size of each task is 64. The learning rates of meta-train and meta-test are $\alpha = 0.001$ and $\beta = 0.01$. For level-aware learning, the number of iterations, learning rate and batch size in Step II are 30, 0.001 and 32. The temperature and weighted hyper-parameter of contrastive learning loss are $\tau = 0.07$ and $\lambda = 0.5$.

We use the model with the highest average QWK on all attributes in validation set for test set and report the average result across five random seeds. All experimental results in this paper are obtained from a Nvidia[4] GeForce RTX 3080 graphics card.

### 4.3. Baseline Models

The details of baseline models are as follows:

(1) **Hi att** (Dong et al., 2017) is a hierarchical structure with an attention-based model. We use the same structure in the encoder in this paper.

(2) **AES aug** (Hussein et al., 2020) is a multi-attribute scoring model based on the structure proposed by Taghipour and Ng (2016).

(3) **PAES** (Ridley et al., 2020) employs handcrafted features to facilitate the acquisition of prompt-agnostic information within a hierarchical structure.

(4) **CTS** (Ridley et al., 2021) marks the inception of cross-prompt multi-attribute scoring, featuring both shared and private layers. This model leverages trait(attribute)-attention mechanism to effectively integrate information from all attributes.

(5) **CTS no att** (Ridley et al., 2021) has the same structure as CTS, with both public and private layers, but excludes trait attention.

(6) **PMAES** (Chen and Li, 2023) proposes a prompt-mapping contrastive learning to learn about more consistent representations from source and target prompts.

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

| Model | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | Avg |
|---|---|---|---|---|---|---|---|---|---|
| PLAES | **0.648** | **0.563** | **0.604** | **0.623** | 0.634 | **0.593** | **0.403** | **0.533** | **0.575** |
| w/o PG | 0.632 | 0.519 | 0.602 | 0.611 | 0.627 | 0.577 | 0.382 | 0.451 | 0.550 |
| w/o LA | 0.604 | 0.535 | 0.542 | 0.604 | 0.621 | 0.512 | 0.395 | 0.494 | 0.538 |
| w/o PG+LA | 0.598 | 0.533 | 0.564 | 0.620 | **0.635** | 0.576 | 0.388 | 0.372 | 0.536 |

Table 4: Ablation results in average QWK of all attributes on each prompt. PG is prompt-generalized learning and LA is level-aware learning. w/o PG is only trained with LA and scoring task, w/o LA is model trained with PG in Step I and scoring task in Step II. w/o PG+LA is model only trained with scoring task.

| Model | Overall | Cont | Org | WC | SF | Conv | PA | Lan | Nar | Avg |
|---|---|---|---|---|---|---|---|---|---|---|
| PLAES | **0.673** | **0.574** | **0.491** | **0.579** | **0.580** | **0.447** | **0.601** | **0.554** | **0.631** | **0.570** |
| w/o PG | 0.646 | 0.551 | 0.434 | 0.555 | 0.552 | 0.383 | 0.594 | 0.551 | 0.620 | 0.543 |
| w/o LA | 0.643 | 0.534 | 0.464 | 0.540 | 0.556 | 0.402 | 0.554 | 0.522 | 0.573 | 0.532 |
| w/o PG+LA | 0.634 | 0.541 | 0.416 | 0.506 | 0.526 | 0.349 | 0.583 | 0.547 | 0.617 | 0.524 |

Table 5: Ablation results in average QWK for each attribute over all prompts.

## 5. Results and Analysis

### 5.1. Main Results

Following Ridley et al. (2021), we report the result from two dimensions. For the first dimension, we show the scoring performance of the models on each prompt in Table 2. We can see that PLAES achieves the best results on all but P1 and gets the average QWK of 0.575. For each attribute dimension over all prompts, we show the scoring performance of the models on each attribute on Table 3. In this dimension, the results demonstrate that our approach PLAES not only achieves the best average QWK (0.570). We also perform a statistical experiment using the pairwise t-test. The results show that PLAES is statistically significant in comparing with PMAES for both the average QWK of all attributes on each prompt (with $p = 0.0177$) and the average QWK for each attribute over all prompts (with $p = 0.0172$).

Among these models, Hi att and AES aug only use neural features based on scoring model. PAES, CTS no att and CTS use additional handcrafted features on the basis of neural networks. The above methods do not consider using other constraints to enhance the generalization ability of the model. PMAES uses contrastive learning to make the model learn more shared features from consistent representations. This strategy is similar to our proposed prompt-generalized learning, which also aims to obtain more general knowledge across source prompts, but we do not need to use any target prompt essay. Meanwhile, our method further utilizes level-aware learning to improve model's capacity to differentiate essays quality within the constraints of writing level. This also enables our method to achieve better performance.

### 5.2. Ablation Studies

In order to investigate the effectiveness of our proposed prompt-generalized learning (PG) and level-aware learning (LA) strategies, we conduct ablation studies in this section. We also present the ablation results from two dimensions.

As shown in Table 4, we can see that removing PG or LA results in a performance decrease on each prompt. The average QWK drops by 2.5% after removing PG, by 3.7% after removing LA, and by 3.9% after removing both PG and LA. From Table 5, we also can seen that either removing PG or LA results in a decrease in the model's scoring performance on all attributes. The above results demonstrate the good effectiveness of our proposed two learning strategies. In conclusion, it can be observed that both PG and LA contribute to the improvement of scoring performance, and LA plays a greater role in improving scoring performance. The performance of our model (full PLAES) is further enhanced by combining PG and LA.

### 5.3. Visualization Analysis

In this section, we use the t-SNE toolkit (Van der Maaten and Hinton, 2008) to visualize essay representations to demonstrate the effects of prompt-generalized learning (PG) and level-aware learning (LA) strategies, respectively.

#### 5.3.1. Effect of Prompt-generalized Learning

As shown in Figure 3, the essay representations encoded by randomly initialized (RI) are significantly inconsistent. After training with PG (RI-PG), the overlap between the source prompts substantially increased, indicating an improvement in prompt representation consistency. The findings indicate that

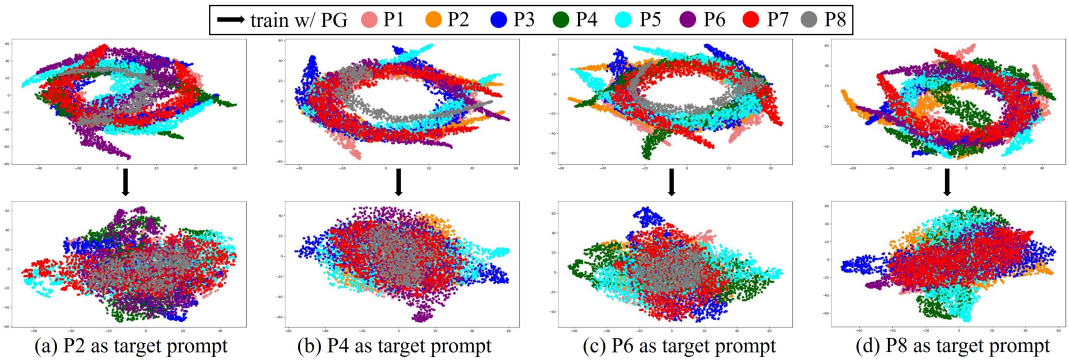


Figure 3: Visualization of source prompt essay representations encoded by randomly initialized models (called RI, top row) and the models after training PG (called RI-PG, bottom row).

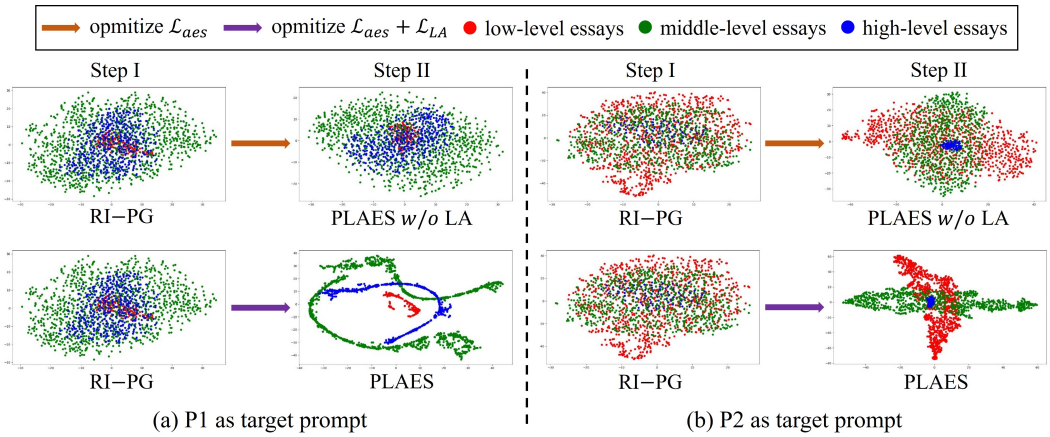


Figure 4: Visualization of target prompt essay representations encoded by RI-PG, PLAES w/o LA and PLAES. Essays in low, middle, and high levels are represented by the red, green, and blue points.

the representations acquired by the RI-PG exhibit greater consistency compared to those obtained by RI alone. Additionally, PG demonstrates the ability to effectively capture more generalized features across different source prompts, which can be used for level-aware learning.

#### 5.3.2. Effect of Level-aware Learning

As shown in Figure 4 (a), when the P1 as the target prompt, if the model only trained the scoring task in Step II by PLAES w/o LA, the model cannot distinguish the three writing levels in the target prompt. This can be reflected that these three level essay representations obtained by PLAES w/o LA is still chaotic. Conversely, after training with LA, the essay representations of three levels became visually distinguishable, indicating that the model has better ability to distinguish between different writing levels of target prompt. The same observation can also be seen in Figure 4 (b). The above results show that LA can effectively improve model's ability to distinguish between different writing levels.

| L-M / M-H | P1 | P3 | P5 | P6 | Avg |
|---|---|---|---|---|---|
| 0.4 / 0.8 | 0.648 | 0.604 | 0.634 | 0.593 | 0.620 |
| 0.2 / 0.8 | 0.635 | 0.595 | 0.617 | 0.580 | 0.607 |
| 0.4 / 0.6 | 0.619 | 0.593 | 0.610 | 0.584 | 0.602 |

Table 6: Experiment results of different boundaries for essay levels. L-M denotes the boundary between low and middle levels, while M-H is the boundary between middle and high levels.

### 5.4. Analysis of Essay Level Boundaries

We conduct experiments on two different boundaries to explore the impact of essay level boundaries on level-aware learning. As shown in Table 6, reducing the low to middle level boundary from 0.4 to 0.2 or the middle to high level boundary from 0.8 to 0.6 will result in a decline in the model's scoring performance. This suggests that in level-aware learning, the score range of low-level may need to be expanded to encompass more essays with lower scores as low-level essays, while the score range of high-level needs to be narrowed to maintain the quality of high-level essays.

| Model | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | Avg |
|---|---|---|---|---|---|---|---|---|---|
| PLAES (32) | **0.648** | **0.563** | **0.604** | 0.623 | **0.634** | **0.593** | **0.403** | **0.533** | **0.575** |
| PG-SupCL (32) | 0.573 | 0.497 | 0.549 | 0.583 | 0.594 | 0.551 | 0.389 | 0.463 | 0.525 |
| PG-SupCL (64) | 0.550 | 0.514 | 0.551 | 0.618 | 0.593 | 0.549 | **0.403** | 0.461 | 0.530 |
| PG-SupCL (128) | 0.577 | 0.505 | 0.549 | **0.624** | 0.607 | 0.560 | 0.386 | 0.467 | 0.534 |
| PG-SupCL (256) | 0.582 | 0.488 | 0.551 | 0.619 | 0.594 | 0.530 | 0.367 | 0.458 | 0.524 |
| PLAES $w/o$ PG (32) | **0.632** | **0.519** | **0.602** | 0.611 | 0.627 | **0.577** | 0.382 | 0.451 | **0.550** |
| SupCL (32) | 0.570 | 0.492 | 0.585 | 0.620 | 0.631 | 0.564 | **0.385** | 0.426 | 0.534 |
| SupCL (64) | 0.564 | 0.489 | 0.588 | **0.624** | 0.622 | 0.562 | 0.371 | 0.459 | 0.535 |
| SupCL (128) | 0.583 | 0.501 | 0.596 | 0.603 | 0.630 | 0.568 | 0.374 | **0.468** | 0.540 |
| SupCL (256) | 0.573 | 0.481 | 0.577 | 0.606 | **0.636** | 0.567 | 0.352 | 0.448 | 0.530 |

Table 7: Comparative experiment of LA and SupCL in average QWK of all attributes on each prompt. $(\cdot)$ presents the batch size when training LA or SupCL.

| Model | Overall | Cont | Org | WC | SF | Conv | PA | Lan | Nar | Avg |
|---|---|---|---|---|---|---|---|---|---|---|
| PLAES (32) | **0.673** | **0.574** | **0.491** | **0.579** | **0.580** | **0.447** | **0.601** | **0.554** | **0.631** | **0.570** |
| PG-SupCL (32) | 0.601 | 0.527 | 0.452 | 0.531 | 0.532 | 0.390 | 0.557 | 0.501 | 0.578 | 0.519 |
| PG-SupCL (64) | 0.601 | 0.534 | 0.450 | 0.532 | 0.529 | 0.389 | 0.572 | 0.510 | 0.590 | 0.523 |
| PG-SupCL (128) | 0.616 | 0.535 | 0.444 | 0.542 | 0.534 | 0.380 | 0.578 | 0.525 | 0.599 | 0.528 |
| PG-SupCL (256) | 0.610 | 0.520 | 0.433 | 0.538 | 0.527 | 0.369 | 0.571 | 0.505 | 0.589 | 0.518 |
| PLAES $w/o$ PG (32) | **0.646** | **0.551** | 0.434 | **0.555** | **0.552** | **0.383** | **0.594** | **0.551** | 0.620 | **0.543** |
| SupCL (32) | 0.593 | 0.538 | **0.441** | 0.533 | 0.522 | 0.365 | 0.592 | 0.544 | 0.621 | 0.528 |
| SupCL (64) | 0.598 | 0.538 | 0.438 | 0.536 | 0.518 | 0.372 | 0.591 | 0.546 | **0.626** | 0.529 |
| SupCL (128) | 0.623 | 0.540 | 0.435 | 0.540 | 0.535 | 0.379 | 0.585 | 0.540 | 0.625 | 0.534 |
| SupCL (256) | 0.614 | 0.525 | 0.409 | 0.531 | 0.531 | 0.350 | 0.586 | 0.549 | 0.618 | 0.524 |

Table 8: Comparative experiment of LA and SupCL in average QWK for each attribute over all prompts.

### 5.5. Level-aware Learning vs SupCL

Level-aware learning is to improve the model's ability to distinguish writing levels. There are many ways to achieve this, such as supervised contrastive learning (Khosla et al., 2020) (SupCL), which pull closer the samples from same category in the same batch and keep away from the samples with different categories. We argue that SupCL may be suboptimal for distinguishing different levels in cross-prompt AES. To verify this, we conduct experiments to replace the LA with SupCL, where the model with PG in Step I is called PG-SupCL, and the model without PG is called SupCL. We keep other settings the same as PLAES.

As shown in Table 7 and Table 8, when LA is replaced by SupCL with the batch size 32, the average QWK of these two dimensions are significantly dropping. The results show that the LA is more effective than SupCL in cross-prompt AES. Furthermore, we found that the number of middle-level essays is significantly larger than that of low-level and high-level essays. When a batch contains almost all middle-level essays, it may lead to unstable model training. Therefore, we also conduct experiments with different batch sizes. We can see that the average QWK of PG-SupCL improves with increase of batch size and get the best QWK when batch size is 128. But this is still far from the result of PLAES. Similar results can be found for PLAES $w/o$ PG and SupCL.

To sum up, we can obtain the following findings: 1) SupCL is affected by the batch size. Too large or too small batches can affect the model's scoring performance. 2) PG and LA can promote each other, but PG and SupCL are incompatible. 3) Compared to SupCL, LA can improve the scoring performance more effectively for cross-prompt AES.

## 6. Conclusion

In order to capture more general knowledge across prompts and improve the model's capacity to differentiate essay quality under the constraint of writing levels, we propose a prompt-generalized and level-aware learning framework for cross-prompt AES. Experiments on the ASAP++ dataset illustrate that our approach outperforms all baseline models. Ablation results show that both prompt-generalized and level-aware learning strategies are effective improving model's scoring performance. In the future, more good strategies can be used for learning general knowledge and more constraints are needed to help the model learn consistent scoring knowledge.

## 7. Acknowledgements

This work is supported by the National Natural Science Foundation of China [grant number: 61976062].

## 8. Bibliographical References

Yigal Attali and Jill Burstein. 2006. Automated essay scoring with e-rater® v. 2. *The Journal of Technology, Learning and Assessment*, 4(3).

Yue Cao, Hanqi Jin, Xiaojun Wan, and Zhiwei Yu. 2020. Domain-adaptive neural automated essay scoring. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1011–1020.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Yuan Chen and Xia Li. 2023. PMAES: Prompt-mapping contrastive learning for cross-prompt automated essay scoring. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1489–1503.

Ronan Cummins, Meng Zhang, and Ted Briscoe. 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)*, pages 789–799.

Yann Dauphin, Harm De Vries, and Yoshua Bengio. 2015. Equilibrated adaptive learning rates for non-convex optimization. *Advances in neural information processing systems*, 28.

Yuning Ding, Marie Bexte, and Andrea Horbach. 2023. Score it all together: A multi-task learning study on automatic scoring of argumentative essays. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13052–13063.

Heejin Do, Yunsu Kim, and Gary Geunbae Lee. 2023. Prompt- and trait relation-aware cross-prompt essay trait scoring. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1538–1551.

Fei Dong and Yue Zhang. 2016. Automatic features for essay scoring - an empirical study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1072–1077. The Association for Computational Linguistics.

Fei Dong, Yue Zhang, and Jie Yang. 2017. Attention-based recurrent convolutional neural network for automatic essay scoring. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), Vancouver, Canada, August 3-4, 2017*, pages 153–162. Association for Computational Linguistics.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910.

Xu Han, Yuqi Luo, Weize Chen, Zhiyuan Liu, Maosong Sun, Zhou Botong, Hao Fei, and Suncong Zheng. 2022. Cross-lingual contrastive learning for fine-grained entity typing for low-resource languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2241–2250.

Yaqiong He, Feng Jiang, Xiaomin Chu, and Peifeng Li. 2022. Automated chinese essay scoring from multiple traits. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12-17, 2022*, pages 3007–3016.

Marti A Hearst. 2000. The debate on automated essay grading. *IEEE Intelligent Systems and their Applications*, 15(5):22–37.

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

Mohamed A Hussein, Hesham A Hassan, and Mohammad Nassef. 2020. A trait-based deep learning automated essay scoring system with adaptive feedback. *International Journal of Advanced Computer Science and Applications*, 11(5).

Ting Jiang, Jian Jiao, Shaohan Huang, Zihan Zhang, Deqing Wang, Fuzhen Zhuang, Furu Wei, Haizhen Huang, Denvy Deng, and Qi Zhang. 2022. PromptBERT: Improving BERT sentence embeddings with prompts. In *Proceedings of the*

*2022 Conference on Empirical Methods in Natural Language Processing*, pages 8826–8837.

Zhiwei Jiang, Tianyi Gao, Yafeng Yin, Meng Liu, Hua Yu, Zifeng Cheng, and Qing Gu. 2023. Improving domain generalization for prompt-aware essay scoring via disentangled representation learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12456–12470.

Cancan Jin, Ben He, Kai Hui, and Le Sun. 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)*, pages 1088–1097.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Linjun Li, Tao Jin, Xize Cheng, Ye Wang, Wang Lin, Rongjie Huang, and Zhou Zhao. 2023. Contrastive token-wise meta-learning for unseen performer visual temporal-aligned translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10993–11007.

Xia Li, Minping Chen, and Jian-Yun Nie. 2020. SEDNN: shared and enhanced deep neural network model for cross-prompt automated essay scoring. *Knowledge-Based Systems*, 210:106491.

Shining Liang, Linjun Shou, Jian Pei, Ming Gong, Wanli Zuo, Xianglin Zuo, and Daxin Jiang. 2022. Label-aware multi-level contrastive learning for cross-lingual spoken language understanding. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9903–9918.

Dongliang Liao, Jin Xu, Gongfu Li, and Yiru Wang. 2021. Hierarchical coherence modeling for document quality assessment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13353–13361.

Fei Mi, Minlie Huang, Jiyong Zhang, and Boi Faltings. 2019. Meta-learning for low-resource natural language generation in task-oriented dialogue systems. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 3151–3157.

Michael Mohler and Rada Mihalcea. 2009. Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 567–575.

Qiong Nan, Danding Wang, Yongchun Zhu, Qiang Sheng, Yuhui Shi, Juan Cao, and Jintao Li. 2022. Improving fake news detection of influential domain via domain-and instance-level transfer. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2834–2848.

Isaac Persing and Vincent Ng. 2013. Modeling thesis clarity in student essays. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 260–269.

Peter Phandi, Kian Ming A Chai, and Hwee Tou Ng. 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*, pages 431–439.

Robert Ridley, Liang He, Xin-yu Dai, Shujian Huang, and Jiajun Chen. 2021. Automated cross-prompt scoring of essay traits. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 13745–13753.

Robert Ridley, Liang He, Xinyu Dai, Shujian Huang, and Jiajun Chen. 2020. Prompt agnostic essay scorer: A domain generalization approach to cross-prompt automated essay scoring. *arXiv preprint arXiv:2008.01441*.

Lawrence M Rudner and Tahung Liang. 2002. Automated essay scoring using bayes' theorem. *The Journal of Technology, Learning and Assessment*, 1(2).

Yafet Salim, Valdi Stevanus, Edwardo Barlian, Azani Cempaka Sari, and Derwin Suhartono. 2019. Automated english digital essay grader using machine learning. In *2019 IEEE International Conference on Engineering, Technology and Education (TALE)*, pages 1–6. IEEE.

Takumi Shibata and Masaki Uto. 2022. Analytic automated essay scoring based on deep neural networks integrating multidimensional item response theory. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2917–2926.

Wei Song, Kai Zhang, Ruiji Fu, Lizhen Liu, Ting Liu, and Miaomiao Cheng. 2020. Multi-stage pre-training for automated chinese essay scoring. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6723–6733.

Md Arafat Sultan, Cristobal Salazar, and Tamara Sumner. 2016. Fast and easy short answer grading with high accuracy. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1070–1075.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112.

Kaveh Taghipour and Hwee Tou Ng. 2016. A neural approach to automated essay scoring. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 1882–1891.

Yi Tay, Minh Phan, Luu Anh Tuan, and Siu Cheung Hui. 2018. Skipflow: Incorporating neural coherence features for end-to-end automatic text scoring. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Masaki Uto, Yikuan Xie, and Maomi Ueno. 2020. Neural automated essay scoring incorporating handcrafted features. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6077–6088.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Cong Wang, Zhiwei Jiang, Yafeng Yin, Zifeng Cheng, Shiping Ge, and Qing Gu. 2023. Aggregating multiple heuristic signals as supervision for unsupervised automated essay scoring. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13999–14013.

Dong Wang, Ning Ding, Piji Li, and Haitao Zheng. 2021. Cline: Contrastive learning with semantic negative examples for natural language understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2332–2342.

Yongjie Wang, Chuang Wang, Ruobing Li, and Hui Lin. 2022. On the use of bert for automated essay scoring: Joint learning of multi-scale essay representation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3416–3425.

Sara Cushing Weigle. 2002. *Assessing writing*. Cambridge University Press.

Xing Wu, Chaochen Gao, Liangjun Zang, Jizhong Han, Zhongyuan Wang, and Songlin Hu. 2022. Esimcse: Enhanced sample building method for contrastive learning of unsupervised sentence embedding. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3898–3907.

Jiayi Xie, Kaiwei Cai, Li Kong, Junsheng Zhou, and Weiguang Qu. 2022. Automated essay scoring via pairwise contrastive regression. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2724–2733.

Ming Yan, Hao Zhang, Di Jin, and Joey Tianyi Zhou. 2020. Multi-source meta transfer for low resource multiple-choice question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7331–7341.

Huaxiu Yao, Ying-xin Wu, Maruan Al-Shedivat, and Eric Xing. 2021. Knowledge-aware meta-learning for low-resource text classification. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1814–1821.

Yuhao Zhang, Hongji Zhu, Yongliang Wang, Nan Xu, Xiaobo Li, and Binqiang Zhao. 2022. A contrastive framework for learning sentence representations from pairwise and triple-wise perspective in angular space. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4892–4903.

## 9. Language Resource References

Sandeep Mathias and Pushpak Bhattacharyya. 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)*.