Increasing the Generalizability of Similarity-Based Essay Scoring Through Cross-Prompt Training

Marie Bexte[1] **and Yuning Ding**[1] **and Andrea Horbach**[1, 2, 3]
[1]CATALPA, FernUniversität in Hagen, Germany
[2]IPN - Leibniz Institute for Science and Mathematics Education, Kiel, Germany
[3]University of Kiel, Germany

Abstract

In this paper, we address generic essay scoring, i.e., the use of training data from one writing task to score data from a different task. We approach this by generalizing a similarity-based essay scoring method (Xie et al., 2022) to learning from texts that are written in response to a mixture of different prompts. In our experiments, we compare within-prompt and cross-prompt performance on two large datasets (ASAP and PERSUADE). We combine different amounts of prompts in the training data and show that our generalized method substantially improves cross-prompt performance, especially when an increasing number of prompts is used to form the training data. In the most extreme case, this leads to more than double the performance, increasing QWK from .26 to .55.

1 Introduction

In automated scoring, one desideratum is often to train a generic classifier that does not rely on the availability of training material for a certain writing task, i.e., prompt, but can transfer from training material for one or several prompts to data from new writing tasks.

This holds both for content scoring, also known as short-answer scoring, and essay scoring. In content scoring, texts of up to a few sentences in length are scored for conceptual correctness. Essay scoring deals with scoring longer texts that are rated both on content and language use.

Generic scoring has a high practical relevance in the classroom, as teachers often do not have the resources to annotate training data for each new prompt. However, the generalizability of classifiers is often low (see, e.g., Phandi et al. (2015)). Especially in a hard domain transfer scenario when classifiers are trained on a single or a few prompts only, they might pick up on lexical material specific to that particular writing task.

For instance, as shown on the left side of Figure 1, two essays from the prompt 'The Face on Mars' in the PERSUADE dataset may lead a scoring classifier trained solely on this prompt to treat words such as 'aliens' and 'Mars' as significant features. These words, however, are not found in essays from other prompts, such as the two essays from the 'Facial Action Coding System' prompt shown on the right side. Despite the differences in content, essays from different prompts with the same score share general similarities. For instance, low-scoring essays from different prompts (top part of Figure 1) often share weaknesses such as limited vocabulary, repetition of phrases, and overuse of simple words. In contrast, high-scoring essays (bottom part of Figure 1) display features that contribute to higher scores, such as a logical progression with the underlined transitional phrases. These lexical patterns should be prioritized when training a generic scoring model, as they contribute significantly to the overall quality of an essay, regardless of the specific prompt. However, it should be noted that we are not claiming that these elements are the *only* relevant aspects in scoring the data, but rather that they are important *enough* to make them exploitable for cross-prompt scoring.

While generic scoring has been more extensively explored for some content scoring datasets (Bailey and Meurers, 2008; Mohler and Mihalcea, 2009; Meurers et al., 2011; Dzikovska et al., 2013), cross-prompt approaches to essay scoring have only received more interest in recent years (Phandi et al., 2015; Jin et al., 2018; Li et al., 2020; Chen and Li, 2023).

In our study, we approach generic essay scoring by training classifiers that are discouraged from paying attention to prompt-specific material in the essays. In both flavors of educational free-text scoring, content and essay scoring, similarity-based scoring has recently emerged as a viable alternative to the default of instance-based scoring (Bexte

Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications, pages 225–236

Prompt: The Face on Mars

I think that if we ever did find **aliens** on **Mars** or even a civilazation we should stay away from it. The reason why is because if we ever did find a other planet we never really be called the only planet in the universe. In the article we found out that we have a new way to find people. We never really thought that their was ever a civilazation on **Mars**. … I have said that we should never really go to planet Mars because if we did we would be the only planet in the universe. This article was very interesting, but I think we should never go near **Mars**.

ID: 5CC530A8C191, Score: 1

Since we had sent surveyors to **Mars** we have found many interesting things. … The face that we have discovered is not an **alien** structure because almost no true scientists think it is alien, high resolution cameras showed no signs of **alien** life, and it is natural because it is in a mesa biom.
The first reason that you should not believe that the face is an **alien** monument is because almost all the scientists agree that it is not a **alien** structure. As it said in the article, "Few scientists believed the Face was an **alien** artifact."…
The second reason is that the first image was burry and did not have the best quality. …
The biggest and most obvious reason that should convince everyone is that it is a mesa biom. …
There is no proof that **aliens** are on **Mars**. There is no proof that **aliens** are real. …

ID: D9C24EDF97A9, Score: 5

Prompt: Facial Action Coding System

i don't think its a bad way to test it i say they should test it to one **classroom** see how it works see what it can do and stuff like that so if you ever want to put theses in **classroom's** give it to one class to try them out and if they like it use but if they don't u can't do anything but make something better or something just like give it a diffenert name and shape, and size stuff like that beacuse you don't want to give it to a person that dose not like the idea of using it. … so i think it would be great to have in **classrooms** as long as they like them and know's how to use them is the biggest part of this new product.

ID: 2C4BB3ACA451, Score: 1

The **Facial Action Coding System** is a very useful tool used to help tell what other peoples emotions are just by their facial expresions. Having this technology can be very helpful in the classrom. It can help define the mixed emotions of a student, how they feel of a topic they are learning, and if a student is telling the truth.
The first reason as to why this technology is useful in the **classroom** is defineing the mixed emotions of students. In the artical it says "The software can even identify mixed emotions" (paragraph 4). …
Another reason this would be useful in class is by defining how they feel about a topic in class. …
The final reason **Facial Action Coding Systems** should be in **classrooms** is because it can help tell when a student is lying. …
In conclusion, this technology would be very useful for **classrooms** to define a students emotions to improve work, help a confused or bored student, and spot out lairs.

ID: 11F4FD622FD8, Score: 5

Figure 1: Example essays taken from two different prompts in the PERSUADE dataset that share the same low (top) or high (bottom) score. Words in bold are prompt-specific, which may be picked up by a classifier trained on a single prompt. The underlined transitional phrases show an example of lexical patterns that contribute to higher scores, which can be used when training a generic scoring model.[1]

et al., 2022; Xie et al., 2022). While instance-based scoring learns the association between individual learner texts and their scores, the input in similarity-based scoring are pairs of texts. In such a pair, an essay of interest is compared to a reference essay with a known score.

We adapt the similarity-based essay scoring approach of Xie et al. (2022), which exhibits state-of-the-art performance on the commonly used ASAP essay scoring dataset. While Xie et al. (2022) only demonstrated good within-prompt performance, we augment their approach for cross-prompt scoring. Our crucial step in avoiding overfitting to prompt-specific information is to only use pairs of learner essays that answer different writing prompts during training. In doing this, we force the similarity metric to pay attention to structural rather than purely lexical similarity between texts.

We hypothesize that the problem of prompt-specific similarity metrics is more severe in cases where training material only covers a single or a few prompts, as paying attention to a prompt-specific feature makes an impact on a larger portion of the dataset in these cases. To test this assumption, we vary the number of prompts that is mixed in the training data in our experiments.

Overall, our paper makes the following contributions:

- We extend the method of Xie et al. (2022) to facilitate cross-prompt scoring.
- We compare two strategies to pair up training data in similarity-based cross-prompt scoring.
- We demonstrate the benefits of our strategy for increasing cross-prompt performance on two publicly available datasets (PERSUADE and ASAP), finding that the benefits of our method increase when an increasing number of prompts is mixed in the training data.

Our code and data split is available on GitHub[2].

2 Related Work

For many years, the main interest in automated essay scoring has been in prompt-specific classifiers, where one specific model was trained for each new

[2] https://github.com/mariebexte/generalizing-similarity

[2] In this example, we use the first two prompts from the dataset, which happen to include the words 'face' and 'facial'. While these shared terms might influence a general classifier trained specifically on these prompts, this is merely a coincidence and not the intended focus of our analysis.

writing prompt (e.g., Taghipour and Ng (2016); Dong et al. (2017); Dasgupta et al. (2018); Uto et al. (2020)). This focus has shifted to generic or cross-prompt scoring, where a classifier is trained on one or more prompts. The classifier is then applied to essays that answer prompts which were not seen during training.

2.1 Cross-Prompt Essay Scoring

The problem of cross-prompt essay scoring has been approached in various ways. Phandi et al. (2015) use Bayesian Linear Ridge Regression to score essays using features selected to be predictive of either the source or the target domain. Jin et al. (2018) propose a two-stage neural network (TDNN) approach, in which they use a generic model to automatically create pseudo-training data for the target domain. Li et al. (2020) also propose a two-stage method that aims to extract the shared knowledge between the source and target domain, first creating pseudo-training data, which is then used in a Siamese network. The PMAES system (Chen and Li, 2023) uses a prompt-mapping contrastive learning method to learn more consistent representations of source and target prompts. By doing this, unlabeled data from the target prompt is used to adapt the model. Thus, adaptation to future target prompts would require additional training. Similarly, Zhang et al. (2025) and Wang et al. (2025) also include information derived from unlabeled target data in their training.

2.2 Similarity-Based Essay Scoring

Orthogonal to cross-prompt scoring, recent years have also seen more and more approaches that rely on the similarity between text pairs for scoring instead of training a classifier on features extracted from individual texts (see also Horbach and Zesch (2019)).

The purported advantage that similarity-based approaches might work better in a cross-domain scenario has been refuted, at least for content scoring (Bexte et al., 2023). However, little work so far has explored the potential of cross-prompt similarity-based essay scoring.

3 Method

In a similarity-based scoring setup, the predicted score is derived from a comparison with reference essays. We follow the prompt-specific approach of Xie et al. (2022), which essentially predicts how much better or worse than a reference essay an essay of interest is. Figure 2 shows an overview of the network structure of this approach. In practice, training essays are used as reference essays, i.e., training is performed on pairs of training essays, and at inference, validation or test essays are compared to training essays. While Xie et al. (2022) use a BERT (Devlin et al., 2019) model at the core of their model, we use a Longformer (Beltagy et al., 2020) instead. This is done to accommodate the longer text length typically encountered in essay scoring. We use the *longformer_base_4096* model as provided on Hugging Face[3]. Both the answer of interest and a reference answer are embedded using the same Longformer model. The difference between the two embeddings is subsequently fed into a linear layer, which performs a regression. The aim is to predict the difference in the score of the essay of interest and the reference essay. While the approach is a regression at its core, scores are scaled back to their target ranges upon prediction.

For example, if a zero-point essay was compared to a two-point reference essay, the model should output a score difference of minus two. While the original authors only compare test essays to reference essays that do not share the same score, we refrain from doing this, as we feel it is inappropriate to incorporate knowledge of the true scores of test instances into the pairing strategy.

Xie et al. (2022) demonstrate that their model has good within-prompt performance, i.e., when training a dedicated model for each prompt. We build on this and expand the approach to also allow for cross-prompt scoring. With this augmentation, one can even combine prompts that do not share the same label range. To achieve this, we carefully scale labels and model outputs. An overview of this scaling is given in Figure 5 in the Appendix.

During training, the true labels Y of individual essays are transformed to scaled labels Y_s, so that each $y_s \in Y_s$ is in the range of $[0, 1]$. Note that this scaling takes the prompt an essay belongs to into account, which means that each $y \in Y$ is scaled according to the label range of the prompt the essay belongs to. When pairing up essays to form training pairs, their target label is the score difference of the essays, i.e., their scores are subtracted. Thus, the score difference d_p of a pair will be in the range of $[-1, 1]$, because $d_p = y_i - y_j$ for $y_i, y_j \in Y_s$.

[3] https://huggingface.co/allenai/longformer-base-4096

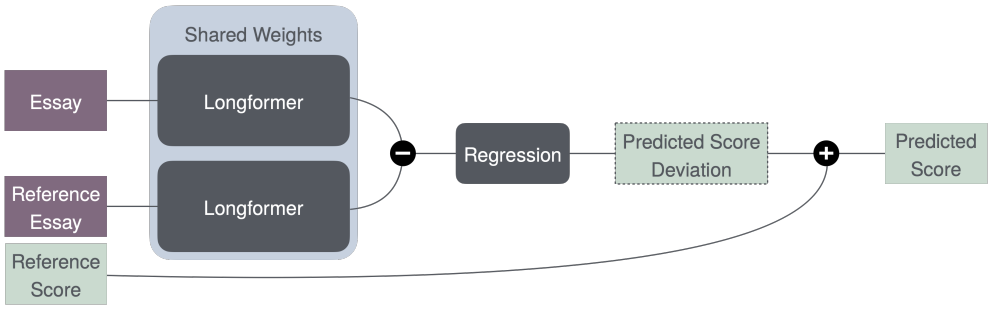

Figure 2: Overview of the model architecture, derived from Xie et al. (2022).

For optimal suitability to the regression model, we again scale the score differences to the range $[0, 1]$.

At inference, validation/test essays are paired up with training, i.e., reference essays. For each test essay t and a reference essay r, the score s_r of r is first scaled to the score range $S_t = [s_{min}, s_{max}]$ of t for compatibility. In processing the pair of t and r, the model outputs the predicted score deviation $\hat{d}$, which lies in the range $[0, 1]$. This now has to be mapped to the label range of t. However, because $\hat{d}$ is a *deviation*, it has to be scaled to the range $[s_{min} - s_{max}, s_{max} - s_{min}]$, which represents the minimal and maximal deviation that is possible within S_t. The result $\hat{d}_t$ can then be used to obtain the predicted score $\hat{s}_t$ of t by adding the predicted score deviation to the true score s_r of r, i.e. $\hat{s}_t = s_r + \hat{d}_t$.

To investigate whether we can nudge the model towards learning less prompt-dependent representations, we contrast two ways of pairing essays during training: In the **standard** setting, we only pair essays from the same prompt. In our **generalize** setting, we only pair essays from different prompts.

Note that our main motivation is to evaluate the effect of building cross-prompt training pairs, rather than to achieve the best possible performance. In the interest of saving energy and time, we thus set our hyperparameters somewhat lower than Xie et al. (2022) did. We always train for five (as opposed to 80) epochs, taking the model with the best performance on the validation data. At inference, we limit ourselves to comparing each validation (testing) essay to 15 (25) training essays (as opposed to 50). The average predicted score is then taken as the final prediction of the model. Although our switch to a Longformer instead of the smaller BERT model increases runtime, we do not make use of the full length of 4,096 tokens. Instead, we truncate inputs to a length of 1,024, as the majority of essays fits in this length[4]. Just like Xie et al. (2022), we use a batch size of 6, and a learning rate of 1e-4.

	PERSUADE	ASAP
# integrated prompts	7	4
# independent prompts	8	4
avg. # essays per prompt	1,733	1,622
avg. essay length in tokens	410.96	222.74
score range	1-6	prompt -dependent

Table 1: Key statistics of the two datasets used in our study.

4 Data

We work on two different data sets, **PERSUADE** and ASAP-aes (Automated Student Assessment Prize - Automatic Essay Scoring), which we refer to as **ASAP**. The core statistics for each dataset can be found in Table 1. Although PERSUADE is best-suited for our analysis due to the large number of prompts, we additionally run our experiments on ASAP, as this is a commonly used essay scoring dataset.

4.1 PERSUADE

The PERSUADE dataset (Crossley et al., 2024) comprises seven integrated prompts, with a total of 12,875 essays written by students from the 6th to the 10th grade, and eight independent prompts with a total of 13,121 essays sampled from writers from the the 8th to the 12th grade. While integrated prompts refer to some source material, independent prompts do not. Each essay was annotated with a holistic score by two raters. Scores range from 1.0 to 6.0 in increments of 1.0. The raters were trained on a standardized SAT holistic essay

[4] 3% of PERSUADE essays and 2.7% of ASAP essays are truncated due to this.

scoring rubric for the independent essays[5] and its modified version for the integrated essays[6]. The main difference between the two rubrics is that the one for the integrated prompts mentions having to include evidence from the reading text[7]. Due to this explicit inclusion of the source text in the rubric, we expect the cross-prompt transfer to be more successful for the independent prompts. Overall, raters showed a strong agreement (weighted $\kappa = .74$) in annotating the essays.

4.2 ASAP

The ASAP dataset[8] is one of the benchmark datasets for automated essay scoring. It contains four integrated and four independent (persuasive/narrative/expository) tasks, spanning a total of 12,978 essays. The essays were written by students from the 7th to the 10th grade. ASAP prompts have also been scored holistically but using a wide variety of different scales. Each essay was evaluated by two raters, with an inter-annotator agreement of $\kappa = .55$. After adjudication, the resulting score ranges can span as little as four or up to 61 different labels, as can be seen in Table 5 in the Appendix. This label incompatibility between prompts further complicates cross-prompt scoring.

5 Experimental Study

In the following, we first describe the overall setup and then present the results of our similarity-based cross-prompt scoring on the two datasets. Our experiments ran on Nvidia Quadro RTX 6000, A40, and A6000 GPUs for around 550 hours.

5.1 Experimental Setup

Our overall goal is to train an essay-scoring classifier that focuses on general indicators of a good essay as opposed to overly relying on prompt-specific features. In our similarity-based method, we facilitate this through the selection of training pairs. We contrast the performance of models trained using pairs that consist of two answers to the same vs. different prompts.

Data Split For each of our datasets, we sample the same number of answers for each prompt, downsampling to the number of answers of the prompt with the lowest answer count. In doing this, we randomly sample a subset of 1,000 essays for each of the 15 prompts in the PERSUADE dataset. 800 of these are used for training and 100 for validation and testing each. For each of the eight prompts in the ASAP dataset, we randomly sample a subset of 700 essays. 560 of these are used for training and 70 for validation and testing each.

In similarity-based scoring, the training data pool is used to build pairs of instances. We derive our strategy to build these pairs from Xie et al. (2022) but relax it to allow data from multiple prompts to be paired. Their strategy includes dropping training pairs of essays with the same score, which we in preliminary experiments found to be a reasonable step, as it cut training time at a minor performance loss.[9] However, we have to ensure that each run, i.e., all combinations of different prompts we use in our experiments, uses the same number of training pairs. Otherwise, runs with more pairs may have a performance advantage. We thus pre-calculate the maximum number of pairs we can build in each of our runs: We determine how many pairs we would end up with if we paired up all essays in the training data that do not share the same score. We then take the minimum of this as the number of training pairs we build in our experiments. This results in 1,495 training pairs for PERSUADE and 920 training pairs for ASAP.

As mentioned earlier, we limit the number of pairs during validation (testing) to 15 (25) pairs per essay. The pairing strategy for the validation data reflects the training setting: If training is done on pairs of essays from the same prompt, validation instances are also paired with training essays from the same prompt. If training is done on pairs of essays from different prompts, we also pair validation essays with training essays from a different prompt. The pairing strategy during testing is 'greedy' in the sense that we check whether essays from the same prompt appeared in the training data. If this is the case, we use 25 of these as reference answers, otherwise, we randomly take 25 essays from the training pool as reference answers.

Single-Prompt Baseline As a starting point for our experiments regarding the impact of training

[5] https://github.com/scrosseye/persuade_corpus_2.0/blob/main/sat_rubric_only_indy.pdf

[6] https://github.com/scrosseye/persuade_corpus_2.0/blob/main/sat_rubric_only_source_based.pdf

[7] The reading texts were not published, which is why we are unable to include them in our analyses.

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

[9] Note that this only applies to the *training* process. As we remark in Section 3, we do not look at scores when building pairs for *test* instances, since we feel that this incorporation of knowledge about scores would be inappropriate.

on combinations of data from multiple prompts, we train models on **single prompts**.

To compare the performance of the similarity-based approach, we also train an **instance-based** classifier. For this instance-based classification, we use the same *longformer_base_4096* model that is also at the heart of the similarity-based approach and attach a classification head. In both instance-based and similarity-based training, we use the same data splits, but for the instance-based classification we adapt the labels of the ASAP dataset to allow for cross-prompt evaluation. To unify the differing label ranges of the ASAP prompts, scores are scaled into a range from 0 to 3, which corresponds to the smallest label range present in the dataset[10]. Models are trained for 10 epochs with a maximum input length of 1,024 tokens, a learning rate of 1e-5, and a batch size of 2.

Mixed-Prompt Scoring Setup We compare models trained on answer pairs from the same prompt to models that were trained with pairs of answers to different prompts.

We vary how many prompts are combined in the training data and hypothesize that combining more prompts leads to a better generalizability of the classifier, i.e., a better cross-prompt performance. To ensure comparability, we keep the overall number of training instances constant for all combinations. The validation data is composed of the same prompts that appear in the training data to make the transfer to the prompts in the test data a hard one. Just as for the training data, the amount of validation data is also downsampled to keep it at the same overall number of instances as when data from a single prompt is used.

For PERSUADE, we report individual results for the seven integrated and eight independent prompts, and for combinations of all 15 prompts. As ASAP only comprises a total of eight prompts (4 integrated, 4 independent), we do not perform a separation into integrated and independent prompts for this dataset and only report results for the gradual combination of all eight prompts. Whenever there are more than ten possible combinations of prompts (e.g., there are 70 ways of picking four out of the eight independent PERSUADE prompts), we randomly sample ten combinations to cut training time, making sure that each prompt was selected in at least one combination.

Evaluation We always evaluate in two different conditions: **within-prompt**, which comprises the test data splits for all prompts that also appear in the training data for that run, and **cross-prompt**, which comprises the test data splits of all other prompts. We expect an increasing number of prompts mixed in the training data to have different effects for the two training and evaluation conditions. Overall, within-prompt evaluation should perform better than cross-prompt evaluation. For cross-prompt evaluation, we expect the generalized training to outperform the standard training. When evaluating in the within-prompt condition, the expectation would be for the models obtained with standard training to outperform those resulting from generalized training, as the former are more attuned to prompt-specific information.

The metric we use to evaluate model performance is quadratically weighted kappa (QWK; Cohen (1968)). Whenever we average QWK results, we perform Fisher Z-transformation to stabilize the variance.

5.2 Results: Single-Prompt Training

Before reporting the results of training on combinations of prompts, we first establish the performance level achieved by training on a single prompt. These results are shown in Table 2.

It is expected that a model will perform best when trained exclusively on data from the same prompt it is later evaluated on. This could thus be seen as somewhat of an upper bound. Table 2 also contains cross-prompt performance, first on all cross-prompt test data and then separated into integrated and independent prompts. We observe that for both PERSUADE and ASAP alike, there is a clear drop in the performance of cross-prompt compared to within-prompt evaluation.

In the case of PERSUADE, models trained on integrated prompts fare similarly in the cross-prompt evaluation, irrespective of whether the test prompts are integrated or independent. However, for models trained on independent prompts, cross-prompt evaluation within the same group (i.e., on another independent prompt) shows an average improvement of 0.16 QWK compared to evaluation on an integrated prompt. This pattern differs for ASAP, perhaps due to the widely varying scoring ranges. Here, the performance of evaluating on integrated vs. independent prompts is similar for models trained on

[10]Note that this is not necessary for the similarity-based scoring, as this method comes with the capability to internally scale prompts with different label ranges into compatibility.

Train	Within-Prompt	Cross-Prompt		
		All		
		PERSUADE		
0	.76	.62	.56	.66
1	.85	.66	.64	.67
2	.66	.41	.40	.41
3	.75	.60	.64	.56
4	.72	.53	.61	.46
5	.69	.63	.61	.64
6	.71	.53	.58	.48
7	.80	.60	.52	.67
8	.81	.65	.57	.73
9	.82	.62	.52	.70
10	.67	.59	.52	.65
11	.74	.66	.61	.71
12	.73	.55	.47	.63
13	.83	.69	.60	.75
14	.68	.56	.45	.64
Avg.	.74	.57	.58	.56
Avg.	.77	.62	.53	.69
Avg.	.75	.60	.56	.63
		ASAP		
3	.70	.37	.36	.38
4	.80	.39	.42	.36
5	.79	.53	.49	.56
6	.81	.53	.71	.36
1	.79	.40	.40	.39
2	.70	.48	.48	.47
7	.81	.49	.53	.45
8	.75	.30	.18	.44
Avg.	.78	.46	.51	.42
Avg.	.76	.42	.40	.44
Avg.	.77	.44	.46	.43

Table 2: QWK performance of models trained on single prompts. Results distinguish integrated and independent prompts. The mapping of prompt numbers to names in PERSUADE is listed in Table 4 in the Appendix.

an independent prompt, but we see a benefit when models trained on an integrated prompt are evaluated on a different integrated prompt.

Comparison to Instance-Based Scoring We further examine the validity of the similarity-based approach by comparing it to a standard instance-based setting. Table 3 compares the average performance of the similarity-based (taken from Table 2) and the instance-based approach. The two setups perform on par on PERSUADE, and similarity-based scoring even outperforms the instance-based classification on ASAP.

5.3 Results: Training on Multiple Prompts

Figure 3 shows the results for training on a mix of different prompts. The number of prompts in the training data gradually increases from left to right. Note that curves start with the results from the previous experiment, where we only trained on a single prompt. A constant benefit of building cross-prompt as opposed to within-prompt training pairs can be observed: The performance of models trained using within-prompt training pairs (dotted lines) tends to drop off, while models trained on cross-prompt training pairs (solid lines) tend to remain more stable or even increase in performance. Contrary to our hypothesis, training on cross-prompt pairs even consistently leads to better performance than standard training for the within-prompt evaluation, thus showing that this training setup does no harm but instead benefits performance across the board.

Strikingly, from a mixture of five prompts onward, our generalization-focused models perform better in cross-prompt evaluation (solid line with crosses) than the standard (i.e., within-prompt-trained) models on within-prompt data (dotted lines with dots) on the ASAP data. We see the same result from a mix of six prompts onward for the longer PERSUADE curves. With the shorter PERSUADE curves, the two conditions again meet at the mark of combining five prompts but remain on a similar performance level from there on. Thus, when five or more prompts are combined, the generalized training strategy pushes cross-prompt performance above standard within-prompt training and evaluation.

6 Embedding Space Analysis

To gain an understanding of how the embedding space is affected by either training exclusively on within-prompt or cross-prompt training pairs, Figure 4 shows embedding space visualizations. To produce these visualizations, we embed the respective test data using the Longformer model that is at the core of the model pipeline. We then use t-SNE to bring the embeddings into 2D space. For t-SNE, we use the sklearn (Pedregosa et al., 2011) implementation at its default values. From the distributions of essay embeddings, one can gather that the models trained using cross-prompt training pairs produce embeddings that are less separated into individual prompts, indicating that they truly learned a more generic representation of the essays.

7 Conclusion and Outlook

Our baseline results confirm the overall solid performance of the model, in line with what Xie et al. (2022) found. In addition, our results demonstrate

	PERSUADE								ASAP							
	Instance-based				Similarity-based (ours)				Instance-based				Similarity-based (ours)			
	Within-Prompt	Cross-Prompt All			Within-Prompt	Cross-Prompt All			Within-Prompt	Cross-Prompt All			Within-Prompt	Cross-Prompt All		
Avg.	.76	.57	.59	.55	.74	.57	.58	.56	.80	.40	.56	.26	.78	.46	.51	.42
Avg.	.77	.62	.55	.69	.77	.62	.53	.69	.60	.27	.23	.33	.76	.42	.40	.44
Avg.	.76	.60	.57	.64	.75	.60	.56	.63	.71	.33	.41	.29	.77	.44	.46	.43

Table 3: Comparison of instance-based and similarity-based scoring, split into the two datasets and their integrated and independent prompts. Both methods perform on par, except for similarity-based scoring outperforming instance-based scoring on the independent ASAP prompts.

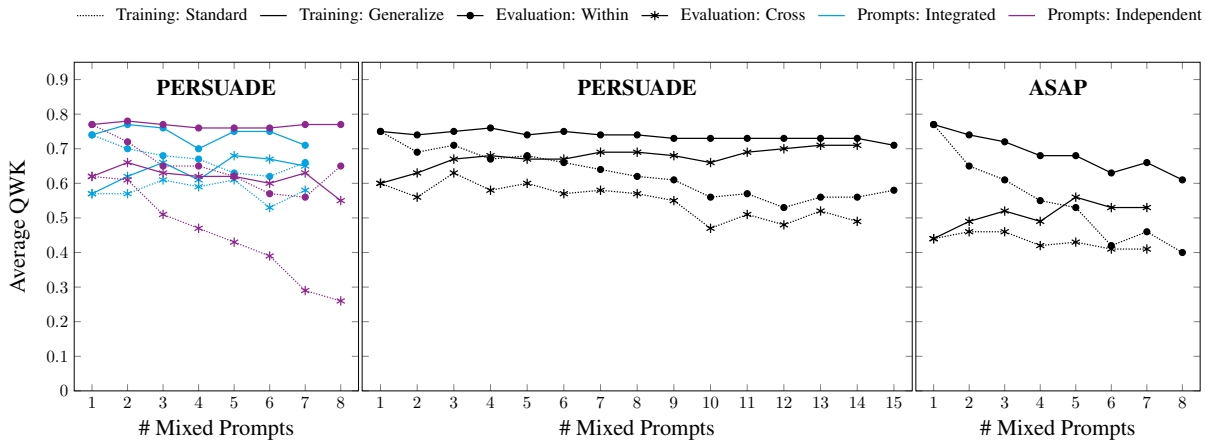

Figure 3: Learning curves depicting how mixing an increasing number of prompts in the training data affects performance. For cross-prompt evaluation, we always use test data from all (i.e., both integrated and independent) prompts that are not in the training data. Our generalized training strategy (solid lines) consistently benefits performance compared to standard training (dotted lines).

the suitability of the model to perform cross-prompt scoring - even in the difficult case of the ASAP dataset with its diverse set of score ranges across different prompts. Our strategy of pairing training essays either within-prompt or cross-prompt proved helpful not only in the cross-prompt scenario but also for within-prompt evaluation. Thus, it is advisable to build training pairs cross-prompt whenever a mixture of multiple prompts is present in the training data.

Limitations and Ethical Considerations

In our setup, we only investigate variants of a hard domain transfer, where data from several source domains is used to train a classifier that is then applied to a target domain. One obvious next step we have not yet taken would be to inject small amounts of target-domain data. Another avenue we do not incorporate is to use the source text of a prompt as a means of facilitating cross-prompt transfer.

Similarly, we do not evaluate cross-prompt performance between datasets. In this study, we restrict ourselves to cross-prompt evaluations within ASAP or PERSUADE (as in almost all related work), i.e., we evaluate on new prompts that are somewhat similar to the source prompts and whose data comes from a similar learner population. The question of the extent to which essay scoring can ever be fully generic remains open and thus requires further research.

As always in automated scoring, fairness and bias are important issues that should be taken into account to make sure that scoring algorithms do not disadvantage certain user groups (see, e.g., Loukina et al. (2019) and Schaller et al. (2024)). These topics also need further investigation for our generic scoring scenario. At the same time, one might argue that a generic classifier is less likely to fall for spurious correlations between scores and unnecessary features than a prompt-specific classifier might be.

Finally, as our experimental setup requires over 1,000 training runs, we make some design choices in the interest of keeping the overall runtime at a reasonable level. Our preliminary results indi-

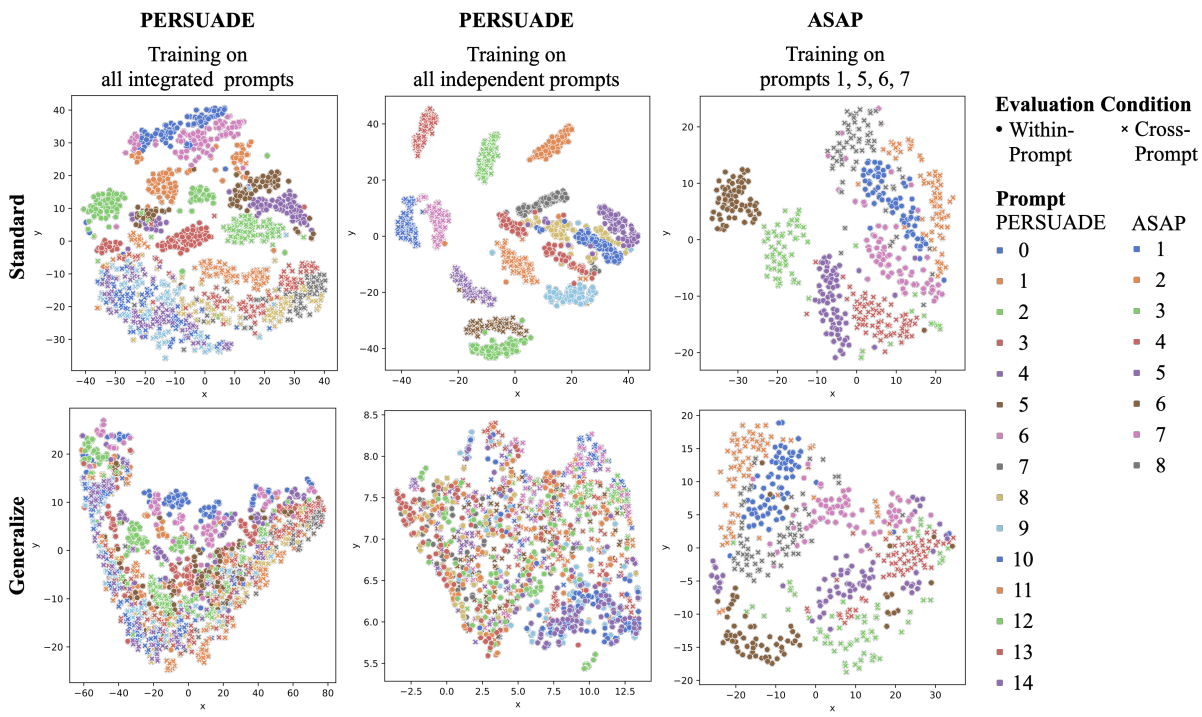

Figure 4: Visualization of embeddings from models trained in the standard (top) and generalize (bottom) conditions, transformed using t-SNE. There is less separation into prompts for models trained with the generalize strategy, indicating that these models do in fact learn a more generalized representation of the essays.

cate that one could achieve better performance than what we report here by training for more than just five epochs, building more training pairs and taking advantage of the full input length of the Longformer model - albeit at the cost of a greater demand on computing resources.

References

Stacey Bailey and Detmar Meurers. 2008. Diagnosing meaning errors in short answers to reading comprehension questions. In *Proceedings of the third workshop on innovative use of NLP for building educational applications*, pages 107–115.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The Long-Document Transformer. *arXiv:2004.05150 [cs]*.

Marie Bexte, Andrea Horbach, and Torsten Zesch. 2022. Similarity-based content scoring - how to make S-BERT keep up with BERT. In *Proceedings of the 17th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2022)*, pages 118–123, Seattle, Washington. Association for Computational Linguistics.

Marie Bexte, Andrea Horbach, and Torsten Zesch. 2023. Similarity-based content scoring-a more classroom-suitable alternative to instance-based scoring? In *Findings of the association for computational linguistics: Acl 2023*, pages 1892–1903.

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, Toronto, Canada. Association for Computational Linguistics.

Jacob Cohen. 1968. Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological Bulletin*, 70(4):213–220. Place: US Publisher: American Psychological Association.

S.A. Crossley, Y. Tian, P. Baffour, A. Franklin, M. Benner, and U. Boser. 2024. A large-scale corpus for assessing written argumentation: PERSUADE 2.0. *Assessing Writing*, 61:100865.

Tirthankar Dasgupta, Abir Naskar, Lipika Dey, and Rupsa Saha. 2018. Augmenting textual qualitative features in deep convolution recurrent neural network for automatic essay scoring. In *Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications*, pages 93–102.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages

4171–4186, Minneapolis, Minnesota. 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)*, pages 153–162.

Myroslava O Dzikovska, Rodney Nielsen, Chris Brew, Claudia Leacock, Danilo Giampiccolo, Luisa Bentivogli, Peter Clark, Ido Dagan, and Hoa Trang Dang. 2013. Semeval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 263–274.

Andrea Horbach and Torsten Zesch. 2019. The influence of variance in learner answers on automatic content scoring. In *Frontiers in education*, volume 4, page 28. Frontiers Media SA.

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.

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.

Anastassia Loukina, Nitin Madnani, and Klaus Zechner. 2019. The many dimensions of algorithmic fairness in educational applications. In *Proceedings of the fourteenth workshop on innovative use of NLP for building educational applications*, pages 1–10.

Detmar Meurers, Ramon Ziai, Niels Ott, and Janina Kopp. 2011. Evaluating answers to reading comprehension questions in context: Results for german and the role of information structure. In *Proceedings of the TextInfer 2011 workshop on textual entailment*, pages 1–9.

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.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

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.

Nils-Jonathan Schaller, Yuning Ding, Andrea Horbach, Jennifer Meyer, and Thorben Jansen. 2024. Fairness in automated essay scoring: A comparative analysis of algorithms on german learner essays from secondary education. In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, pages 210–221.

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.

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.

Jiong Wang, Qing Zhang, Jie Liu, Xiaoyi Wang, Mingying Xu, Liguang Yang, and Jianshe Zhou. 2025. Making meta-learning solve cross-prompt automatic essay scoring. *Expert Systems with Applications*, 272:126710.

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, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Chunyun Zhang, Jiqin Deng, Xiaolin Dong, Hongyan Zhao, Kailin Liu, and Chaoran Cui. 2025. Pairwise dual-level alignment for cross-prompt automated essay scoring. *Expert Systems with Applications*, 265:125924.

A Appendix

This appendix contains supplementary information to increase the transparency and reproducibility of our experiments. Table 4 gives information on the mapping between prompt numbers and names in PERSUADE. For ASAP, Table 5 gives information on the label ranges of the different prompts. To better grasp the generalization of the model for cross-prompt scoring, Figure 5 presents a graphic overview of how labels are scaled during training and inference.

#	Prompt Name
0	The Face on Mars
1	Facial action coding system
2	A Cowboy Who Rode the Waves
3	Does the electoral college work?
4	Car-free cities
5	Driverless cars
6	Exploring Venus
7	Summer projects
8	Mandatory extracurricular activities
9	Cell phones at school
10	Grades for extracurricular activities
11	Seeking multiple opinions
12	Phones and driving
13	Distance learning
14	Community service

Table 4: Prompt mapping in the PERSUADE dataset.

Prompt	Label Range	
	From	To
Integrated Prompts		
3	0	3
4	0	3
5	0	4
6	0	4
Independent Prompts		
1	2	12
2	1	6
7	0	30
8	0	60

Table 5: Label ranges in the ASAP dataset.

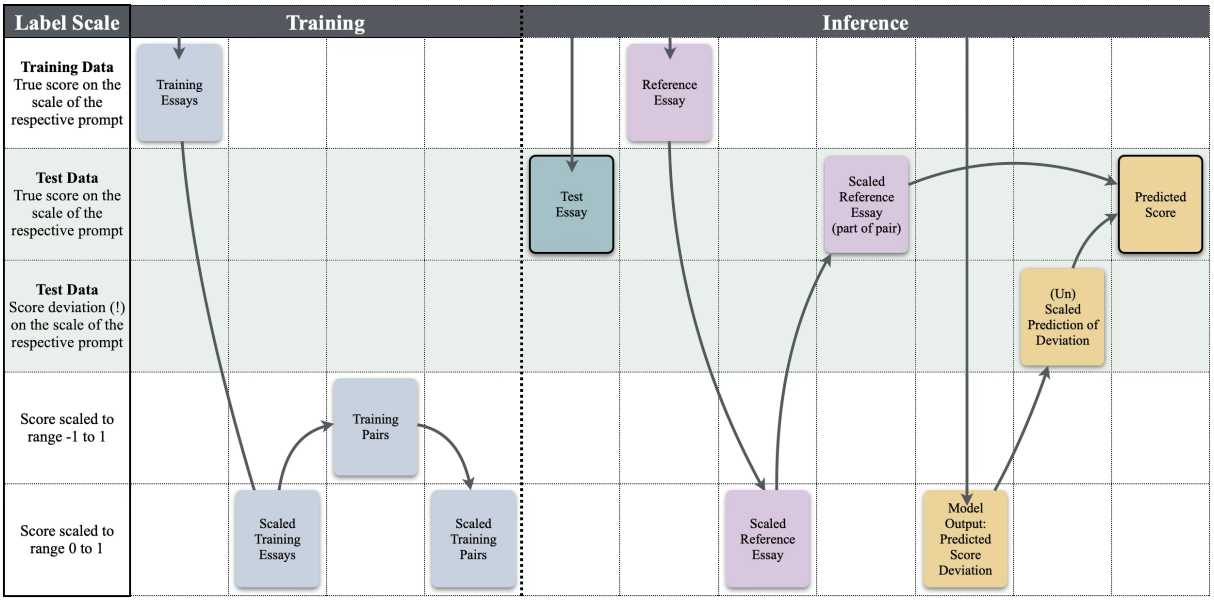

Figure 5: Overview of how labels are scaled to achieve compatibility between score ranges when training on a mix of answers to different prompts.