Automated Essay Scoring Using Grammatical Variety and Errors with Multi-Task Learning and Item Response Theory

Kosuke Doi **Katsuhito Sudoh** **Satoshi Nakamura**
Nara Institute of Science and Technology
{doi.kosuke.de8, sudoh, s-nakamura}@is.naist.jp

Abstract

This study examines the effect of grammatical features in automatic essay scoring (AES). We use two kinds of grammatical features as input to an AES model: (1) grammatical items that writers used correctly in essays, and (2) the number of grammatical errors. Experimental results show that grammatical features improve the performance of AES models that predict the holistic scores of essays. Multi-task learning with the holistic and grammar scores, alongside using grammatical features, resulted in a larger improvement in model performance. We also show that a model using grammar abilities estimated using Item Response Theory (IRT) as the labels for the auxiliary task achieved comparable performance to when we used grammar scores assigned by human raters. In addition, we weight the grammatical features using IRT to consider the difficulty of grammatical items and writers' grammar abilities. We found that weighting grammatical features with the difficulty led to further improvement in performance.[1]

1 Introduction

Automated Essay Scoring (AES) is a task that automatically grades essays. Essay assignments are widely used in language tests and classrooms to assess learners' writing abilities, while grading them takes time and effort for human raters. Maintaining inter- and intra-rater reliability is another issue associated with human scoring. AES can help alleviate these problems and has been attracting more attention in recent years.

The grading methods for essays can be roughly categorized into two types: holistic scoring and analytic scoring. The former assigns a single score to an essay based on its overall performance, while the latter assigns different scores to various aspects of the essay, such as grammar, vocabulary, content, or organization (Weigle, 2002). However, rubrics for holistic scoring typically contain descriptions of several aspects of writing used in analytic scoring (*e.g.*, TOEFL iBT Independent Writing Rubric).

Among those aspects, we focus on grammatical features, inspired by the research on criterial features for the levels of the Common European Frameworks of References (CEFR) (Council of Europe, 2001) in L2 English (Hawkins and Filipović, 2012). The CEFR, one of the influential frameworks in language teaching, describes language abilities in functional terms (*i.e.*, can-do statements, such as "Can write short, simple essays on topics of interest"). However, it is grammatical items and lexis that realize the functions written in can-do statements. To fully develop and elaborate their ideas in essays, they need to use a wide range of grammatical items. In fact, grammar plays an important role in essay scoring. Researches on writing in the second language acquisition field have been focusing on syntactic complexity[2] and accuracy (see Kuiken, 2023; Housen et al., 2012).

Hawkins and Filipović (2012) identified grammatical items that learners at a certain level and higher can use correctly and items that learners at a certain level are prone to making mistakes in. It is known that human raters look for those features consciously or unconsciously when they evaluate learners' performance, and explicit use of grammatical features in AES will improve model performance.

Grammatical features have been used in many feature-engineering AES models (see Ke and Ng, 2019) as well as in hybrid models, which incorporate handcrafted features into deep neural network AES models (Dasgupta et al., 2018; Uto et al., 2020; Bannò and Matassoni, 2022). In

[1]The code is publicly available at https://github.com/ahclab/aes-grammar-mtl-irt.

[2]Syntactic complexity refers to the extent to which a learner can use a wide variety of both basic and sophisticated structures (Wolfe-Quintero et al., 1998).

Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications, pages 316–329

Yannakoudakis et al. (2011), features representing grammatical structures were used together with other linguistic features. However, in many previous studies (*e.g.*, Vajjala, 2018; Uto et al., 2020), grammatical items used correctly were aggregated into measures of grammatical complexity (*e.g.*, ratio of dependent clauses per clauses; see Wolfe-Quintero et al., 1998) rather than individual grammatical items (*e.g.*, adverbial clause *if*, adverbial clause *so that*) even though the difficulties of individual grammatical items are different.

In this paper, we propose to use individual grammatical items as inputs to hybrid AES models that predict holistic scores, and leverage the models to incorporate the variety of grammatical items in grading essays. We also use frequencies of grammatical errors corrected by a modern grammatical error correction model (GECToR-large; Tarnavskyi et al., 2022) as model inputs. The grammatical features are combined with an essay representation and passed into a fully connected feed-forward neural network to predict the score of an input essay. Our models used BERT (Devlin et al., 2019) to learn essay representations following the current state-of-the-art AES models (Yang et al., 2020; Cao et al., 2020; Wang et al., 2022).

To utilize grammatical features more effectively, we develop a multi-task learning framework that jointly learns to predict holistic scores and grammar scores of essays. We use two types of grammar scores: (1) scores assigned to essays by human raters and (2) writers' latent abilities estimated based on patterns of grammar usage using Item Response Theory (IRT) (Lord, 1980). Note that teacher labels are not necessary for estimating the latent abilities using IRT.

IRT estimates not only each writer's ability but also the characteristics of each item (*i.e.*, individual grammatical item), such as discrimination and difficulty parameters. Therefore, we use these IRT parameters to weight grammatical items (*e.g.*, award writers who use a difficult grammatical item; see Section 3.1.2).

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

- We propose to use individual grammatical items and grammatical errors as inputs to AES models, and leverage the models to consider grammar use in predicting holistic scores of essays.
- We develop a multi-task learning framework that jointly learns to predict holistic scores and grammar scores of essays.
- We apply IRT to writers' grammar usage patterns and (1) use estimated latent abilities for multi-task learning, and (2) use IRT parameters to weight grammatical items when we feed them to AES models.
- We show the effectiveness of incorporating grammatical features into BERT-hybrid AES models. Our method shows a significant advantage on some essay assignments in the Automated Student Assessment Prize (ASAP) dataset[3].

2 Related Work

2.1 Automated Essay Scoring

Early AES models predict essay scores using handcrafted features (see Ke and Ng, 2019). For example, e-rater (Burstein et al., 2004) uses 12 features, including grammatical errors and lexical complexity measures. Yannakoudakis et al. (2011) automatically extracted various linguistic features, including grammatical structures, using a parser. These features were weighted and used to train SVM ranking models. Vajjala (2018) reported that measures of grammatical complexity and errors were assigned large weights among various linguistic features.

Recently, a deep neural network-based approach has become popular. AES models based on RNN (Taghipour and Ng, 2016), Bi-LSTM (Alikaniotis et al., 2016), and pre-trained language models (Nadeem et al., 2019; Yang et al., 2020; Cao et al., 2020; Wang et al., 2022) have been proposed. In addition, a hybrid model, which incorporates handcrafted features into a deep neural network-based model, has been proposed (Dasgupta et al., 2018; Uto et al., 2020; Bannò and Matassoni, 2022).

AES using a large language model has also been explored. Mizumoto and Eguchi (2023) demonstrated that using linguistic features in GPT-3 improved AES performance. Yancey et al. (2023) reported that providing a small number of scoring examples to GPT-4 led to comparable performance to models trained on hundreds of thousands of data based on 85 language features.

This study examines the effect of explicitly considering grammatical features in a hybrid AES

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

model by incorporating individual grammatical items as model inputs and weighting them using IRT parameters.

2.2 Multi-Task Learning

Multi-task learning (MTL) (Caruana, 1997) is a method that improves the generalization performance of the main task by training a single model to perform multiple tasks simultaneously. MTL has been used in previous studies in AES, and shown to be effective. Cummins et al. (2016) used MTL to overcome the lack of task-specific data in the ASAP dataset by treating each essay prompt as a different task. Xue et al. (2021) also trained a model jointly on eight different prompts in the ASAP dataset using BERT.

There are also studies that have performed MTL with other NLP tasks. Cummins and Rei (2018) trained an LSTM jointly on grammatical error detection and AES. While the error detection task in Cummins and Rei (2018) required the model to predict whether a particular token was errorful, ones in Elks (2021) require to (1) predict a sentence contains errors or (2) classify tokens by a type of error (*e.g.*, correct, lexical, form). Other auxiliary tasks used in previous studies include morpho-syntactic labeling, language modeling, and native language identification (Craighead et al., 2020), sentiment analysis (Muangkammuen and Fukumoto, 2020), predicting the level of each token (Elks, 2021), and predicting span, type, and quality of argumentative elements (Ding et al., 2023).

In this paper, we train models jointly on holistic scores and grammar scores. This is similar to AES models that predict multiple essay traits simultaneously (Mathias and Bhattacharyya, 2020; Hussein et al., 2020; Mim et al., 2019; Ridley et al., 2021), but the difference between them and ours is that we explicitly incorporate grammatical features to a model, which are related to the score to be predicted.

2.3 Item Response Theory

IRT is a probabilistic model that has been widely used in psychological and educational measurement (Hambleton et al., 1991). An IRT model expresses the probability of a correct response to a test item as a function of the item parameters, which represent the characteristics of the item, and the ability parameter, which represents the ability of the examinee.

Previous studies in AES used IRT to mitigate raters' bias (Uto and Okano, 2021), integrate prediction scores from various AES models (Aomi et al., 2021; Uto et al., 2023), and predict multiple essay traits (Uto, 2021; Shibata and Uto, 2022). These studies employed a multidimensional IRT model since unidimensionality cannot be assumed for the subject to which IRT is applied.

In contrast, we regard individual grammatical items as test items, assuming that whether grammar items are used correctly constitutes grammar ability (*i.e.*, satisfy the assumption of unidimensionality). We model writers' grammar ability using two-parameter logistic model (Lord, 1952), formulated by the following equation:

$$P_{ij}(\theta_i) = \frac{1}{1 + \exp(-Da_j(\theta_i - b_j))} \quad (1)$$

where $P_{ij}(\theta_i)$ is the probability that the writer i with ability θ_i uses the grammatical item j correctly, a_j is the discrimination parameter for item j, and b_j is the difficulty parameter for item j. D is a scaling factor and set to 1.0 in this paper.

3 Proposed Method

3.1 Grammatical Features

The Common European Frameworks of References (CEFR) (Council of Europe, 2001) is an internationally recognized framework for language proficiency. It divides proficiency into six levels ranging from A1 (beginner) to C2 (advanced). Due to the language-neutral nature of the CEFR, what grammatical and lexical properties learners develop across the CEFR levels has been studied language by language.

Such properties (criterial features) in English have been identified by English Profile Programme (Hawkins and Filipović, 2012). Criterial features refer to linguistic properties that are characteristic and indicative of L2 proficiency levels and that distinguish higher levels from lower (*ibid*). They identified positive linguistic features (PFs; grammatical items that learners can use correctly at a certain level and higher) and negative linguistic features (NFs; grammatical items that learners at a certain level are prone to making mistakes in) in relation to the CEFR levels.

Based on the analyses of human raters' grading performance in actual exams, Hawkins and Buttery (2009) have argued that they develop clear intuitions about these properties. We expect that allowing a model to learn grammar representations

Features	Descriptions
type256	256 grammatical items, whether a writer use the items
err54	54 types of errors, # of errors
multiply_b	Modify `type256` with item difficulty
prob	Replace elements in `type256` with the probabilities of using the items correctly
multiply_prob	Weight `type256` with the probabilities
add_prob	Consider both the actual use (`type256`) and the probabilities

Table 1: Grammatical features used in our experiments. The number of errors is relative freq. per 100 words.

using grammatical features would improve the AES performance. Table 1 shows PFs and NFs used in our experiments. The following sections describe them in detail.

3.1.1 Positive Linguistic Features

PFs were extracted using a toolkit for frequency analysis of grammatical items, which is provided by the CEFR-J Grammar Profile (Ishii and Tono, 2018). It extracts 501 grammatical items in text based on regular expressions and calculates the frequencies of them. We converted the frequencies into the 256-dimensional vector (`type256`) based on CEFR-J Grammar Profile for Teachers[4] as $\boldsymbol{g_i} = \{g_{i1}, g_{i2}, ..., g_{i256}\}$. Each dimension corresponds to a grammatical item, and $g_{ij} = 1$ if the writer i used the item j in the essay, and 0 if not.

3.1.2 PFs Weighted using IRT Parameters

Researches on criterial features have shown that learners master more and more grammatical items across the CEFR levels, but `type256` does not consider the difficulties of the items. Therefore, we weight them using the IRT parameters.

We transform g_{ij} in the following four ways:

multiply_b: $g'_{ij} = g_{ij} \times b_j$

prob: $g'_{ij} = P_{ij}(\hat{\theta}_i)$

multiply_prob: $g'_{ij} = g_{ij} \times P_{ij}(\hat{\theta}_i)$

add_prob: $g'_{ij} = \alpha g_{ij} + (1 - \alpha) P_{ij}(\hat{\theta}_i)$

where $\hat{\theta}_i$ is the grammatical ability of the writer i estimated based on the patterns of grammar usage using IRT, and α is a weighting parameter. α was set to 0.5 in this paper.

`multply_b` aims to consider the difficulty of items by multiplying the difficulty parameter for the item. However, writers might not have used some grammatical items because of the essay topic although they had enough abilities to do. Therefore, we use $P_{ij}(\hat{\theta}_i)$, which shows the probability that the writer i with ability $\hat{\theta}_i$ can use the item j correctly. In `prob` g_{ij} is replaced with $P_{ij}(\hat{\theta}_i)$, while in `multiply_prob` and `add_prob` the two values are combined to consider both the ability of writers and the actual use in essays.

IRT parameters were estimated independently from the prediction of essay scores. The IRT parameters were frozen during the training of scoring models.

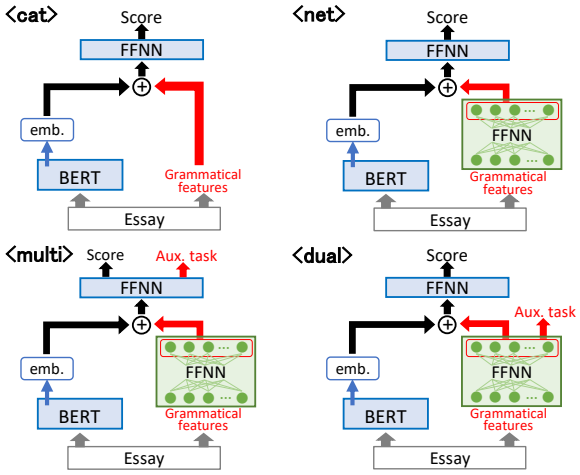

Figure 1: The architectures of proposed models

3.1.3 Negative Linguistic Features

We calculated the number of grammatical errors per 100 words as NFs. Specifically, we created the 54-dimensional vector (`err54`) based on error tags assigned by ERRANT (Bryant et al., 2017)[5]. We used GECToR-large (Tarnavskyi et al., 2022) to correct grammatical errors in essays.

3.2 Model Architecture

Our model takes a batch of essays and grammatical features as input and predicts the holistic scores of the essays. We prepare a model that takes only a batch of essays as input for a baseline. Essay representations are obtained from the `[CLS]` token of the BERT model.

Grammatical features are used in the four settings shown in Figure 1. In `cat`, we concatenate the essay representation and the vector of gram-

[4]https://www.cefr-j.org/download.html#cefrj_grammar The toolkit distinguishes the same items in different sentence types such as the affirmative or negative, while CEFR-J Grammar Profile for Teachers does not.

[5]Based on all possible combinations of the error types and categories. We tried the 24-dimensional vector, which was based on the error types, but the 54-dimensional vector improved the model performance more.

matical features, and feed it to a fully connected feed-forward neural network (FFNN). In net, we first feed the vector of grammatical features to an FFNN and concatenate the representation from the final layer with the essay representation. In multi, we perform multi-task learning with the model architecture of net. The FFNN in multi consists of shared layers only, and does not have task-specific layers[6]. In dual, the predicted values for the auxiliary task are output from the FFNN for grammatical features.

As the labels for the auxiliary task in multi and dual, we used grammar scores assigned to essays by human raters, which is available in ASAP and ASAP++ dataset (Mathias and Bhattacharyya, 2018), and grammar abilities estimated using IRT. Grammar abilities can be estimated from writers' grammar usage patterns without any teacher labels.

4 Experiments

4.1 Data and Evaluation

We used the ASAP and the ASAP++ dataset in our experiments. The ASAP consists of essays for eight different prompts, with holistic scores for Prompts 1-6 and analytic scores for Prompts 7-8. In Prompt 7 and 8, the weighted sum of the analytic scores constitutes the total score, which is the target of prediction by our models. ASAP++ includes analytic scores of essays for Prompt 1-6. We developed AES models that predict the holistic score for each essay prompt. From analytic scores, we only used ones related to grammar[7].

We evaluated the scoring performance of our models using the Quadratic Weighted Kappa (QWK) on the ASAP dataset. Following the previous studies, we adopted 5-fold cross validation with 60/20/20 split for train, development, and test sets, which was provided by Taghipour and Ng (2016).

4.2 Settings

As explained in Section 3.2, we developed our AES models based on BERT. We used bert-base-uncased provided by Hugging Face[8]. The maximum input length was set to 512.

We normalized essay scores in the range of $[-1, 1]$. The mean squared error (MSE) loss was employed for both the main and auxiliary tasks. We updated the parameters for the FFNN and the BERT layers. The number of hidden layers in the FFNN for grammatical features was set to 3, and the number of the nodes in the hidden layer to one-half the dimension of the grammatical features. The number of hidden layers in the FFNN on the top was set to $\{1, 2, 3, 4, 5, 7, 10\}$ for cat, $\{1, 2, 3\}$ for net, and $\{2, 3\}$ for multi and dual and we chose the value that achieved the best QWK score on the development set for Prompt 1. The number of the nodes was set to 512. For both FFNNs, we adopted relu as the activation function and set the dropout ratio to 0.2. In multi and dual, we tried $\{0.8, 0.6\}$ for the weights of the loss function for the main task. We used Adam optimizer (Kingma and Ba, 2015) with a learning rate of 1e-5. We trained models with the batch size $\{4, 8, 16, 32\}$ for 10 epochs. In the following sections, we report the scores on test sets for the batch size with the highest QWK on the development set for each essay prompt. The scores are the average of three experiments with different seed values.

	# of hidden layers						
Model	1	2	3	4	5	7	10
baseline	.813	–	–	–	–	–	–
cat	.792	**.825**	.814	.813	.801	.766	.722
net	.812	**.824**	.817	–	–	–	–
multi–hum (0.8)	–	.819	**.827**	–	–	–	–
multi–hum (0.6)	–	.804	.812	–	–	–	–
dual–hum (0.8)	–	.816	**.824**	–	–	–	–
dual–hum (0.6)	–	.820	.819	–	–	–	–

Table 2: Comparison of the number of hidden layers in FFNN on the top (type256, Prompt 1, QWK dev)

5 Results

5.1 Hyperparameters for Each Model Architecture

Using type256 for the grammar features, we searched for the optimal hyperparameters for each model architecture. Table 2 shows the QWK results on the development set of Prompt 1 when we changed the number of hidden layers in the FFNN on the top. When the number of hidden layers was set to 1 in cat, QWK was lower than the baseline (.792 vs. .813). QWK became the highest when the number of hidden layers was set to 2, while it got lower as the number of hidden layers increased. In net, the architecture with 2 hidden layers achieved the highest QWK. In both multi–hum

[6]We tried models with task-specific layers, but the performance was worse than ones without them.

[7]Conventions for Prompt 1, 2, 7, and 8. Language for Prompt 3-6.

[8]https://github.com/huggingface/transformers

Model	Prompt 1	2	3	4	5	6	7	8	avg.
baseline	.799	.662	.662	.804	.801	.809	.821	.726	.760
+ type256									
cat	.819	.674	.675	.801	.809	.809	.830	.721	.767
net	.814	.679	.678	.810	.806	.806	.831	.737	.770
multi–hum	.816	.678	.683	.812	.810	.811	.830	.746	.773
dual–hum	.818	.673	.687	.819	.807	.813	.833	.750	**.775**

Table 3: Comparison among model architectures (type256, QWK test)

and `dual-hum`[9], QWK became the highest when the number of hidden layers was set to 3 and the weight of the loss for the main task to 0.8. In the subsequent experiments, we trained models using these hyperparameters.

5.2 Comparison among Model Architectures

Using `type256` for the grammar features, we compared the model performance among the four model architectures. Table 3 shows the QWK results on the test set of all prompts. By using `type256`, the average QWK score for all essays improved in all proposed models, compared to the baseline (See avg. in Table 3).

In `cat`, however, the QWK scores did not improve in three prompts (Prompt 4, 6, and 8), which suggests that simple concatenation of essay representations and grammatical features was not sufficient enough to take advantage of the information that the grammatical features have. In `net`, only Prompt 6 did not improve from the baseline, and it seems effective to feed the grammatical features to an FFNN before concatenating with essay representations.

The QWK scores for the models with the auxiliary task (`multi-hum` and `dual-hum`) were higher than the others. Even when looking at the QWK scores for each essay prompt individually, the scores improved for all prompts. These results suggest that multi-task learning with grammar scores is effective to take advantage of grammatical features.

`Dual-hum` achieved the best performance among the four proposed architectures. In `dual-hum`, grammar scores were predicted from the final layer of the FFNN for grammatical features (see Figure 1), which might let the model learn better representations for grammatical features.

Since the `dual-hum` model performed the best, we conducted the subsequent experiments using the setting.

5.3 Comparison of Grammatical Features

Using the `dual-hum` setting, we compared the effectiveness of different grammatical features. Table 4 shows the QWK results on the test sets when we trained models using different grammatical features.[10]

PFs and NFs In the previous section, we showed that positive linguistic features (PFs; `type256`) improved the AES performance. From the Table 4, we can see that negative linguistic features (NFs; `err54`) also improved the model performance (see avg.). Even on a per-prompt basis, the QWK scores were higher for all prompts than those in the baseline.

Combining the PFs and the NFs (`type256 + err54`) also resulted in an improvement in AES performance. However, the average QWK score (.775) was almost same as that for `type256` and `err54`, and no synergistic effect was observed by using both the PFs and the NFs. We just concatenated the vectors of the two features before feeding the features to the FFNN for grammatical features, and there might be more effective ways to combine them.

PFs weighted using IRT parameters We further explored the effectiveness of PFs by weighting them using IRT parameters (see Section 3.1.2). When we considered the difficulties of individual grammatical items (`multiply_b`), the QWK score became the highest among all settings. On the other hand, modifying `type256` with the probability that a writer with a certain grammar ability uses the grammatical item correctly did not help to improve AES performance. Although the QWK scores got higher than that for the baseline, they were lower than that for `type256`. The results suggest that it is more important to capture what items the writer actually used in the essay than what items the writer seemed able to use.

Effect of Grammatical Features To verify that the score improvement came from the addition of grammatical features rather than multi-task learning, we trained models with the auxiliary task but without using grammatical features. The number of hidden layers in the FFNN on the top

[9] "–hum" represents that grammar scores assigned by human raters were used. "–irt" is added when grammar abilities estimated using IRT are used.

[10] The QWK results for the auxiliary task are shown in Appendix A.

Features	Prompt 1	2	3	4	5	6	7	8	avg.
baseline	.799	.662	.662	.804	.801	.809	.821	.726	.760
multi–ffnn1	.803	.680	.659	.797	.802	.806	.827	.723	.762
multi–ffnn3	.812	.671	.684	.812	.805	.812	.831	.748	.772
type256	.818	.673	.687	.819	.807	.813	.833	.750	.775
	(**+.019**)	(+.011)	(**+.025**)	(+.015)	(+.006)	(+.004)	(**+.012**)	(**+.024**)	(+.015)
err54	.815	.672	.689	.813	.805	.812	.832	.756	.774
	(+.016)	(+.010)	(**+.027**)	(+.009)	(+.004)	(+.003)	(+.011)	(**+.030**)	(+.014)
type256+err54	.821	.673	.689	.815	.810	.805	.834	.752	.775
	(**+.022**)	(+.011)	(**+.027**)	(+.011)	(+.009)	(-.004)	(**+.013**)	(**+.026**)	(+.015)
multiply_b	.811	.680	.701	.818	.813	.821	.829	.759	**.779**
	(+.012)	(+.018)	(**+.039**)	(+.014)	(+.012)	(+.012)	(+.008)	(**+.033**)	(+.019)
prob	.820	.661	.682	.813	.807	.808	.834	.752	.772
	(**+.021**)	(-.001)	(**+.020**)	(+.009)	(+.006)	(-.001)	(**+.013**)	(**+.026**)	(+.012)
multiply_prob	.826	.662	.678	.815	.813	.809	.827	.746	.772
	(**+.027**)	(±0)	(+.016)	(+.011)	(+.012)	(±0)	(+.006)	(**+.020**)	(+.012)
add_prob	.812	.674	.682	.806	.799	.812	.827	.757	.771
	(+.013)	(+.012)	(**+.020**)	(+.002)	(-.002)	(+.003)	(+.006)	(**+.031**)	(+.011)
Yang et al. (2020)	.817	**.719**	.698	.845	**.841**	**.847**	.839	.744	**.794**
	(+.017)	(**+.040**)	(+.019)	(**+.023**)	(**+.038**)	(**+.050**)	(+.004)	(+.019)	(**+.026**)
Cao et al. (2020)	.824	.699	**.726**	**.859**	.822	.828	**.840**	.726	.791
	(-.002)	(+.001)	(+.017)	(**+.037**)	(-.002)	(-.001)	(+.011)	(-.017)	(+.006)
Wang et al. (2022)	**.834**	.716	.714	.812	.813	.836	.839	**.766**	.791

Table 4: Comparison among grammatical features (dual–hum, QWK test). The numbers in parentheses indicate the improvement from the baseline. The numbers in parentheses for Yang et al. (2020) and Cao et al. (2020) are the improvement from their baseline, which is equivalent to ours (`RegressionOnly` and `BERT (individual)`, respectively; n/a for Wang et al. (2022)).

was set to 1 (`multi-ffnn1`; same as the baseline) and 3 (`multi-ffnn3`; the best parameter for `multi-hum`; see Section 5.1). The QWK scores for `multi-ffnn1` and `multi-ffnn3` were higher than that of the baseline, but lower than those of the models with grammatical features (Table 4). The results show that both multi-task learning and grammatical features contributed to improve the model performance. In addition, the significant improvement on `multi-ffnn3` suggests that adding layers on the top of BERT would be effective in multi-task learning.

Scoring examples We show some examples from the fold 2 of Prompt 1 (Table 5). The true scores of the four examples are 10, and are written in roughly the same number of words.

In ID 1382, a relatively wide variety of grammatical items were used (10.18 items per 100 words, while the average for essays with true score of 10 included in the fold 2 test set was 9.86). The model trained using `type256` captured the characteristic and predicted the correct score.

Essay ID	# words	Grammatical items # type	Grammatical items per 100	Predicted score baseline	Predicted score type256
1382	442	45	10.18	9	10
377	480	47	9.79	12	11
104	405	38	9.38	9	8
1097	421	42	9.98	9	8

Table 5: Scoring examples. The true scores of the four examples are 10. Per 100 represents the number of different grammar items used per 100 words.

On the other hand, for ID 377 and 104, the model trained using `type256` assigned lower scores than the baseline because of the limited variety of grammatical items in the essays. Note that the prediction improved in ID 377, while it got worse in ID 104.

In ID 1097, our model did not perform well. Although a relatively wide variety of grammatical items were used, the predicted score was lower than that of the baseline.[11]

[11] See Appendix B for the confusion matrix on all the data points.

	Prompt								
Model	1	2	3	4	5	6	7	8	avg.
multi-irt	.819	.669	.697	.811	.813	.821	.839	.757	.778
dual-irt	.805	.678	.686	.807	.808	.816	.831	.742	.772

Table 6: QWK results of the models using the IRT ability parameter for the auxiliary task (QWK test)

Comparison with existing models The QWK scores for the state-of-the-art AES models are also shown in Table 4. The average QWK score of our models (the highest at .779) was not as high as those of the existing models. In some prompts, there seemed to be differences in baseline QWK scores between the previous studies and ours, and we made comparisons based on the improvement from each baseline[12].

In Prompt 1, 3, 7, and 8, our proposed models showed a greater improvement in the QWK scores compared to Yang et al. (2020) and Cao et al. (2020). In these four prompts, the scores themselves of our models were also competitive with those of the existing models. Cao et al. (2020) achieved the state-of-the-art results in Prompt 3, 4, and 7, but the improvements from their baselines were relatively small in the other prompts.

However, our proposed methods were less effective for Prompt 2, 4, 5, and 6, which resulted in lower average QWK scores than the existing models. To identify when the proposed methods were effective, we examined the characteristics of the essays, such as the type of essays, the average number of words in essays, the correlation coefficient between holistic scores and the grammar ability parameter θ and between human-annotated grammar scores and θ, and the variance of θ, but none of them could provide a satisfactory explanation. We need further investigation and it might help to improve the performance on the prompts where our methods were less effective.

5.4 Using the IRT Ability Parameter for the Auxiliary Task

In Section 5.2, we demonstrated that `dual-hum` model achieved the best performance among the four proposed architectures. However, the architecture requires grammar scores annotated by human raters. Therefore we employed grammar abilities estimated using IRT, which requires no human-annotated labels, as the teacher signals.

Table 6 shows that `multi-irt` and `dual-irt` models achieved comparable performance to the models that used human-annotated score. In general, analytical scoring is more time-consuming than holistic scoring, and grammar scores, which are one of the analytical scores, are not always available in a dataset. A method that improves AES performance without the additional human-annotated labels has practical value. Another advantage of using IRT for our AES models is that we can provide the characteristics of grammatical items (*i.e.*, discrimination and difficulty) as well as essay scores.

6 Conclusions

This study examined the effectiveness of using grammatical features in AES models. Specifically, we fed two kinds of features: (1) grammatical items that writers used correctly in essays (PFs), and (2) the number of grammatical errors (NFs). We showed that both PFs and NFs improved the model performance, but combining them did not result in further improvement. The experimental results suggest that multi-task learning would be effective to take advantage of the information that the grammatical features have. One of the future directions could be exploring effective ways to combine PFs and NFs to improve the model performance since the way in this study was a simple concatenation of the two vectors (*e.g.*, to learn representations for PFs and NFs in different networks and combine them). Another direction would be to examine the effectiveness of adding our grammatical features in AES using a large language model. It potentially improves the scoring performance in zero- and/or few-shot settings (Mizumoto and Eguchi, 2023). Furthermore, in order to have more interpretable models, it would be beneficial to analyze how much individual grammatical features contribute to model's score prediction. The insights delivered by interpretable models can help practitioners in education.

We also weighted PFs in several ways using IRT parameters and found that considering the difficulties of grammatical items would improve the model performance. In addition, we used the ability parameter θ as teacher signals for the auxiliary task in multi-task learning. Although no human-annotated labels are required to estimate the IRT

[12]We re-implemented R^2 BERT (Yang et al., 2020), but our re-implementation of the model did not achieve as good scores as those reported in their paper. Furthermore, we trained models using grammatical features with the loss combination proposed by them (*i.e.*, regression and ranking loss), which resulted in lower QWK scores than our baseline.

parameters, the model trained with the ability parameter achieved comparable performance to the model trained with grammar scores annotated by human raters. In this study, IRT parameters were estimated based on grammatical items that writers used in their essays. In the future, we will apply IRT to both PFs and NFs to model writers' grammar abilities.

7 Limitations

Our proposed methods showed significant advantage on some essay prompts in the ASAP dataset, while they were less effective on the other prompts. Further investigation is necessary to clarify what kind of essays our proposed methods would be effective to. An analysis of the effectiveness of grammatical features on different prompts will also provide additional insights into the variation of model behavior across different prompts.

There are also some limitations related to the extraction of grammatical features. First, the toolkit provided by the CEFR-J Grammar Profile extracts grammatical items based on sophisticated regular expression patterns, which was written by a linguist. It would be quite challenging to prepare a similar toolkit in other languages. Bannò and Matassoni (2022) let a model predict the frequencies of grammatical errors from essay representations, which can be applicable to PFs, but the approach requires human-annotated labels to train a model. Another approach is to extract grammatical features based on cross-linguistically consistent annotations such as Universal Dependencies. It makes easier to use grammatical features in other languages, while it remains challenging to extract ones related to parts of speech and/or morphological features rather than dependencies (*e.g.*, present perfect in English).

Second, there could be errors in the extraction using regular expressions and the same is true for grammatical error correction. Experiments using grammatical features annotated by humans would help reveal the influence of errors in feature extraction.

Third, our method requires explicitly extracting grammatical features at test time as well as at training time. An alternative would be to develop a multi-task learning framework where a model is trained to reconstruct grammatical features at training time and then run the trained model on unparsed test data (*e.g.*, Andersen et al., 2021).

Acknowledgments

A part of this work was supported by JSPS KAKENHI Grant Numbers JP21H05054 and by JST, the establishment of university fellowships towards the creation of science technology innovation, Grant Number JPMJFS2137.

References

Dimitrios Alikaniotis, Helen Yannakoudakis, and Marek Rei. 2016. Automatic text scoring using neural networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 715–725, Berlin, Germany. Association for Computational Linguistics.

Øistein E Andersen, Zheng Yuan, Rebecca Watson, and Kevin Yet Fong Cheung. 2021. Benefits of alternative evaluation methods for automated essay scoring. In *Proceedings of the 14th International Conference on Educational Data Mining (EDM21)*, pages 856–864.

Itsuki Aomi, Emiko Tsutsumi, Masaki Uto, and Maomi Ueno. 2021. Integration of automated essay scoring models using item response theory. In *Artificial Intelligence in Education: 22nd International Conference, AIED 2021, Utrecht, The Netherlands, June 14–18, 2021, Proceedings, Part II*, pages 54–59.

Stefano Bannò and Marco Matassoni. 2022. Cross-corpora experiments of automatic proficiency assessment and error detection for spoken English. In *Proceedings of the 17th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2022)*, pages 82–91, Seattle, Washington. Association for Computational Linguistics.

Christopher Bryant, Mariano Felice, and Ted Briscoe. 2017. Automatic annotation and evaluation of error types for grammatical error correction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 793–805, Vancouver, Canada. Association for Computational Linguistics.

Jill Burstein, Martin Chodorow, and Claudia Leacock. 2004. Automated essay evaluation: The criterion online writing service. *AI Magazine*, 25(3):27–36.

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*, page 1011–1020.

Rich Caruana. 1997. Multitask learning. *Machine Learning*, 28:41–75.

Council of Europe. 2001. *Common European Framework of Reference for Languages: Learning, Teaching, Assessment*. Cambridge University Press, Cambridge.

Hannah Craighead, Andrew Caines, Paula Buttery, and Helen Yannakoudakis. 2020. Investigating the effect of auxiliary objectives for the automated grading of learner English speech transcriptions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2258–2269, Online. Association for Computational Linguistics.

Ronan Cummins and Marek Rei. 2018. Neural multi-task learning in automated assessment. *arXiv*, arXiv: 1801.06830.

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, Berlin, Germany. Association for Computational Linguistics.

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, Melbourne, Australia. Association for Computational Linguistics.

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.

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, Toronto, Canada. Association for Computational Linguistics.

Tim Elks. 2021. Using transfer learning to automatically mark L2 writing texts. In *Proceedings of the Student Research Workshop Associated with RANLP 2021*, pages 51–57, Online. INCOMA Ltd.

Ronald K. Hambleton, Hariharan Swaminathan, and H. Jane Rogers. 1991. *Fundamentals of Item Response Theory*. SAGE Publications, California.

John A Hawkins and Paula Buttery. 2009. Using learner language from corpora to profi le levels of profi ciency: insights from the english profi le programme. In Lynda Taylor and Cyril J Weir, editors, *Language Testing Matters Investigating the Wider Social and Educational Impact of Assessment - Proceedings of the ALTE Cambridge Conference April 2008*, pages 158–175. Cambridge University Press.

John A Hawkins and Luna Filipović. 2012. *Criterial Features in L2 English: Specifying the Reference Levels of the Common European Framework*. Cambridge University Press, Cambridge.

Alex Housen, Folkert Kuiken, and Ineke Vedder, editors. 2012. *Dimensions of L2 Performance and Proficiency: Complexity, Accuracy and Fluency in SLA*. John Benjamins, Amsterdam.

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):287–293.

Yasutake Ishii and Yukio Tono. 2018. Investigating japanese EFL learners' overuse/underuse of english grammar categories and their relevance to CEFR levels. In *Proceedings of the 4th Asia Pacific Corpus Linguistics Conference (APCLC 2018)*, pages 160–165.

Zixuan Ke and Vincent Ng. 2019. Automated essay scoring: A survey of the state of the art. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pages 33–40.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations*.

Folkert Kuiken. 2023. Linguistic complexity in second language acquisition. *Linguistics Vanguard*, 9(s1):83–93.

Frederic M. Lord. 1952. *A theory of test scores (Psychometric Monograph No. 7)*. Psychometric Society, Iowa City.

Frederic M. Lord. 1980. *Applications of Item Response Theory To Practical Testing Problems*. Routledge, New York.

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)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Sandeep Mathias and Pushpak Bhattacharyya. 2020. Can neural networks automatically score essay traits? In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 85–91, Seattle, WA, USA → Online. Association for Computational Linguistics.

Farjana Sultana Mim, Naoya Inoue, Paul Reisert, Hiroki Ouchi, and Kentaro Inui. 2019. Unsupervised learning of discourse-aware text representation for essay scoring. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 378–385, Florence, Italy. Association for Computational Linguistics.

Atsushi Mizumoto and Masaki Eguchi. 2023. Exploring the potential of using an ai language model for automated essay scoring. *Research Methods in Applied Linguistics*, 2(2):100050.

Panitan Muangkammuen and Fumiyo Fukumoto. 2020. Multi-task learning for automated essay scoring with sentiment analysis. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing: Student Research Workshop*, pages 116–123, Suzhou, China. Association for Computational Linguistics.

Farah Nadeem, Huy Nguyen, Yang Liu, and Mari Ostendorf. 2019. Automated essay scoring with discourse-aware neural models. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 484–493, Florence, Italy. Association for Computational Linguistics.

Robert Ridley, Liang He, Xin-yu Dai, Shujian Huang, and Jiajun Chen. 2021. Automated cross-prompt scoring of essay traits. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(15):13745–13753.

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

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, Austin, Texas. Association for Computational Linguistics.

Maksym Tarnavskyi, Artem Chernodub, and Kostiantyn Omelianchuk. 2022. Ensembling and knowledge distilling of large sequence taggers for grammatical error correction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3842–3852, Dublin, Ireland. Association for Computational Linguistics.

Masaki Uto. 2021. A multidimensional generalized many-facet rasch model for rubric-based performance assessment. *Behaviormetrika*, 48:425–457.

Masaki Uto, Itsuki Aomi, Emiko Tsutsumi, and Maomi Ueno. 2023. Integration of prediction scores from various automated essay scoring models using item response theory. *IEEE Transactions on Learning Technologies*, 16(6):983–1000.

Masaki Uto and Masashi Okano. 2021. Learning automated essay scoring models using item-response-theory-based scores to decrease effects of rater biases. *IEEE Transactions on Learning Technologies*, 14(6):763–776.

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, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Sowmya Vajjala. 2018. Automated assessment of Non-Native learner essays: Investigating the role of linguistic features. *International Journal of Artificial Intelligence in Education*, 28(1):79–105.

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, Seattle, United States. Association for Computational Linguistics.

Sara Cushing Weigle. 2002. *Assessing Writing*. Cambridge University Press, Cambridge.

Kate Wolfe-Quintero, Shunji Inagaki, and Hae-Young Kim. 1998. *Second language development in writing : measures of fluency, accuracy, & complexity*. University of Hawai'i Press, Honolulu.

Jin Xue, Xiaoyi Tang, and Liyan Zheng. 2021. A hierarchical bert-based transfer learning approach for multi-dimensional essay scoring. *IEEE Access*, 9:125403–125415.

Kevin P. Yancey, Geoffrey Laflair, Anthony Verardi, and Jill Burstein. 2023. Rating short L2 essays on the CEFR scale with GPT-4. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pages 576–584, Toronto, Canada. Association for Computational Linguistics.

Ruosong Yang, Jiannong Cao, Zhiyuan Wen, Youzheng Wu, and Xiaodong He. 2020. Enhancing automated essay scoring performance via fine-tuning pre-trained language models with combination of regression and ranking. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1560–1569, Online. Association for Computational Linguistics.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading ESOL texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 180–189, Portland, Oregon, USA. Association for Computational Linguistics.

A QWK Results for the Auxiliary Task

Table 7 shows the QWK score for the auxiliary task (*i.e.*, predicting grammar score). The QWK scores were generally low, and some of them were negative. We observed that the models output scores

	Prompt								
Model	1	2	3	4	5	6	7	8	avg.
type256	0.032	-0.007	0.050	-0.007	0.001	0.000	0.000	0.079	0.016
err54	-0.002	0.017	-0.003	0.014	-0.007	0.003	-0.012	0.045	0.008
type256+err54	0.148	0.003	0.085	0.001	0.000	0.001	-0.002	0.110	0.039
multiply_b	0.015	-0.003	-0.002	-0.023	0.000	0.000	0.012	-0.003	-0.001
prob	0.052	-0.025	0.028	0.000	0.000	0.000	0.000	0.046	0.008
multiply_prob	0.097	0.000	0.003	0.000	0.000	0.000	0.007	0.059	0.018
add_prob	0.053	-0.023	0.050	-0.002	0.004	0.000	0.039	0.004	0.011

Table 7: QWK results for the auxiliary task on the test set (models shown in Table 4)

close to the mode value in each prompt. One of the possible reasons is the relatively low weight for loss function for the auxiliary task (*i.e.*, 0.2). However, when we assigned a higher weight for the auxiliary task (*i.e.*, 0.4), the model prediction for the main task got worse. Further consideration is necessary for predicting multiple essay traits simultaneously (*e.g.*, Ridley et al., 2021; Shibata and Uto, 2022).

B Detailed Results of Model Predictions

Detailed scoring performance of the model trained using `type256` is shown in Figure 2. The values in the confusion matrices are the sum of all experiments (*i.e.*, 5-fold cross validation and three experiments with different seed values).

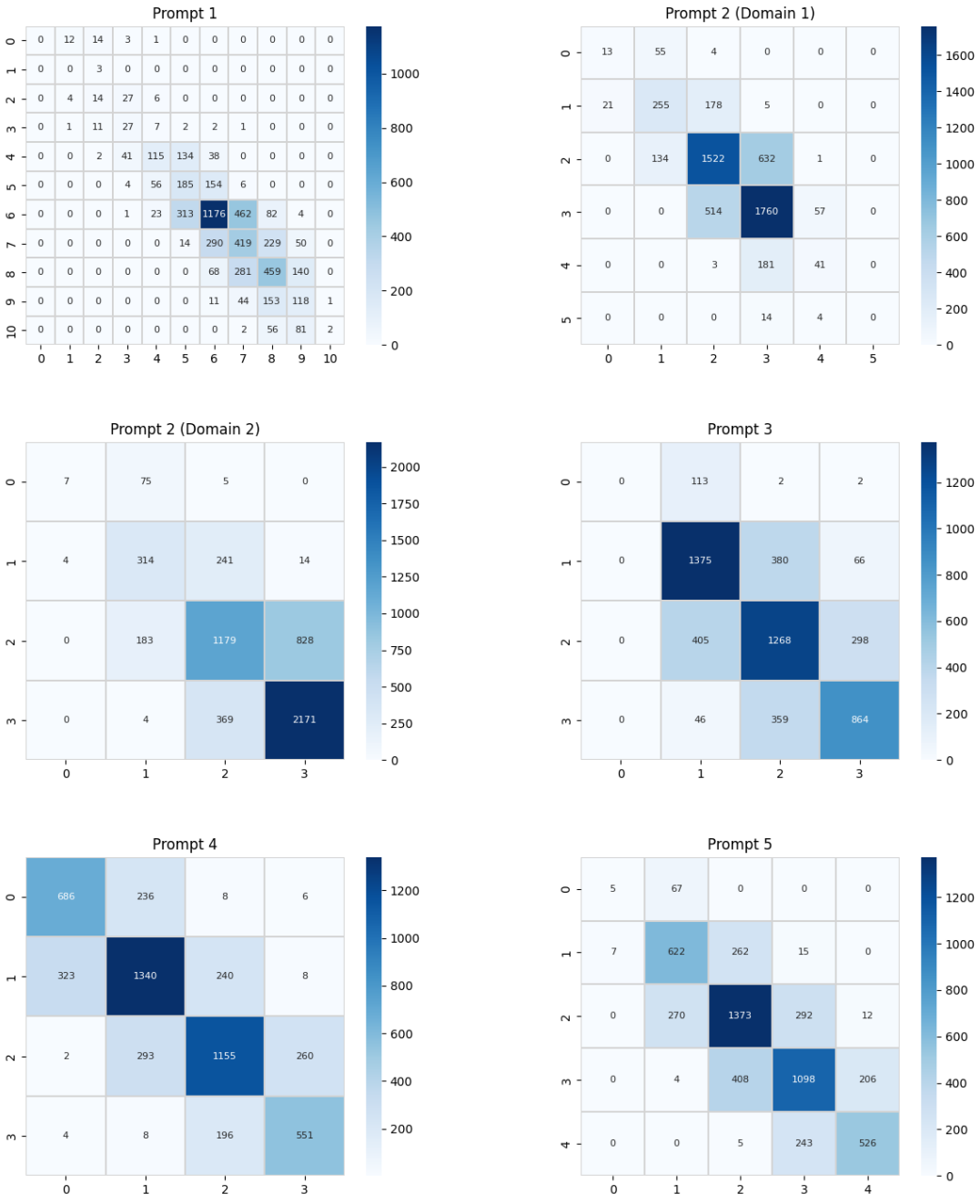

Figure 2: Scoring performance of the model trained using `type256`

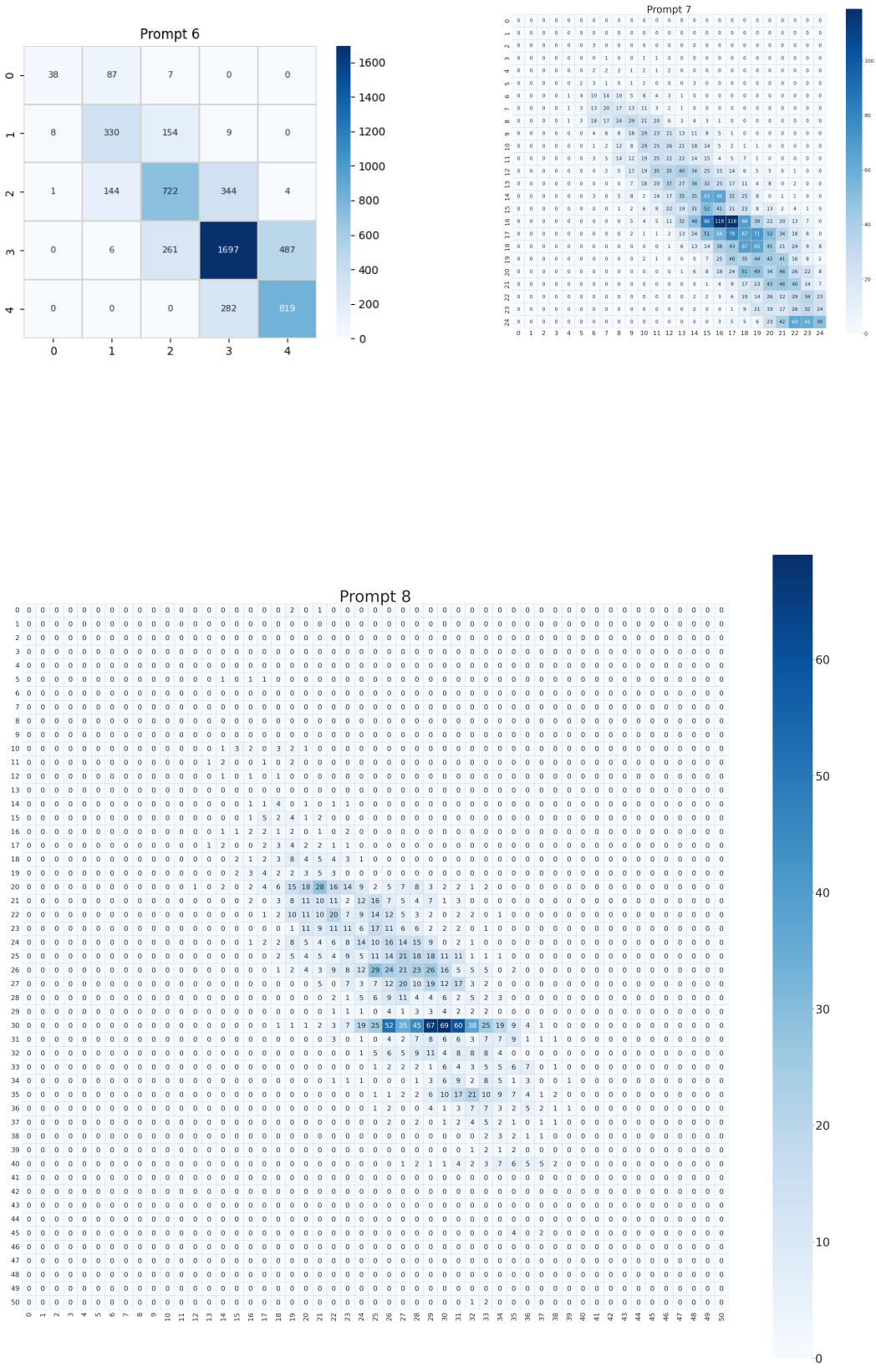

Figure 2: Scoring performance of the model trained using `type256` (*cont.*)