Score It All Together: A Multi-Task Learning Study on Automatic Scoring of Argumentative Essays

Yuning Ding[1], Marie Bexte[1], and Andrea Horbach[1,2]
[1]CATALPA, FernUniversität in Hagen, Germany
[2]Hildesheim University, Germany

Abstract

When scoring argumentative essays in an educational context, not only the presence or absence of certain argumentative elements but also their quality is important. On the recently published student essay dataset PERSUADE, we first show that the automatic scoring of argument quality benefits from additional information about context, writing prompt and argument type. We then explore the different combinations of three tasks: automated span detection, type and quality prediction. Results show that a multi-task learning approach combining the three tasks outperforms sequential approaches that first learn to segment and then predict the quality/type of a segment.

1 Introduction

In educational settings, argumentative essays are considered an important task type, since argumentative writing encourages critical thinking and civic participation (Andrews, 2009), which can only be developed through practice. While **segmentation and classification of argumentative elements** in texts are important steps towards providing learners with feedback on their writing structure, **assessing the quality of arguments** is equally important, but less researched.

For instance, in a writing prompt in the recently published essay corpus PERSUADE (Crossley et al., 2022), students were asked to write a letter to their principal, who is considering making community service a requirement for all students. Two example essays start with the following segments:

(1) *Students all across the nation are doing community service to help out their town or city. Whether it is cleaning the streets, helping out in an old people's home, or tutoring children at a school, all these deeds are good experience and good for the community.*

(2) *I have heard that many students are arguing on whether community service should be done by the students and I just wanted to share my opinions with you to help your decision.*

Both segments are classified as *lead*, an argumentative type that should grab the reader's attention and point toward the writer's position. The first segment is labeled as *effective*, presumably because it captures the reader's interest through listing some activities and effectively shows the author's support for community service. In contrast, likely because it does not point to any position, the second lead has been labeled as *ineffective*. As humans, we can identify these segments and notice their differences easily, but it is unclear how well a machine can locate, classify and predict the quality rating of each argumentative element.

The essay example shown in Figure 1 comes from of the Kaggle competition dataset "Feedback Prize - Predicting Effective Arguments"[1], which is a subset of the aforementioned PERSUADE Corpus. With the public release of this dataset, large-scale argumentative essay data (4192 essays from 15 writing prompts) became publicly available with seven types of argumentative elements being labeled on a three-point quality scale, marking elements as *effective*, *adequate* or *ineffective*. This gives rise to the following research questions:

1. How does the scoring of argument quality interact with additional information in the essay? Is it beneficial to add context, writing prompt and argument type?
2. Can we jointly learn where an argument occurs, which type and of what quality it is? How do the three tasks benefit from each other?

We answer both questions through the following two studies:

[1] https://www.kaggle.com/competitions/feedback-prize-effectiveness

Findings of the Association for Computational Linguistics: ACL 2023, pages 13052–13063
July 9-14, 2023

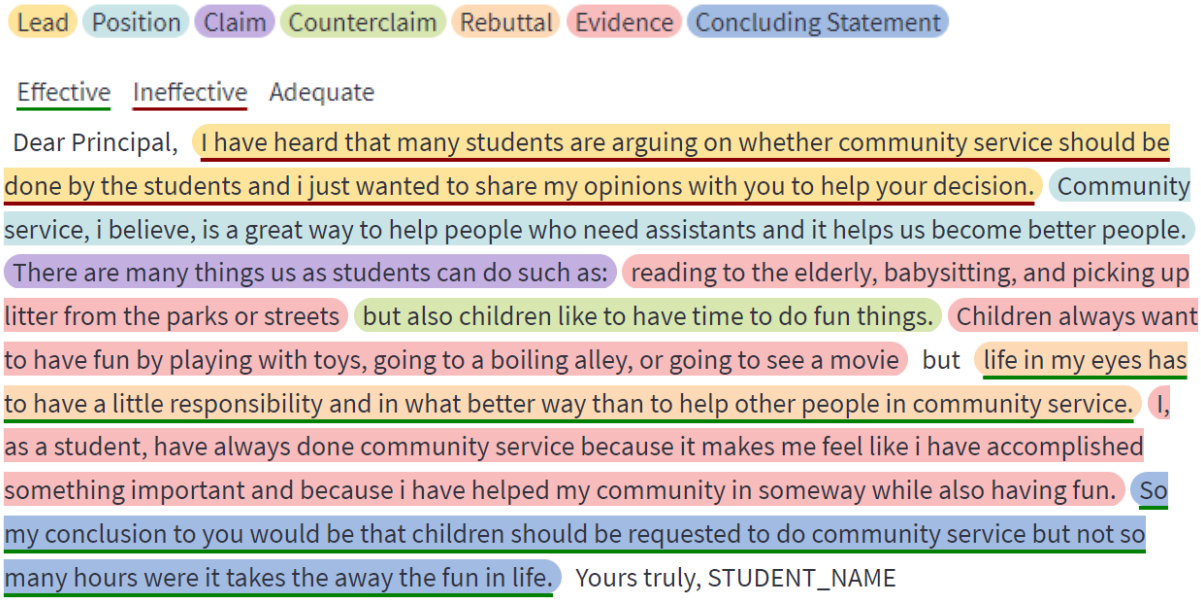

Figure 1: An essay example containing all seven argumentative types and all three quality levels.

In Study 1, we treat the quality prediction of arguments as a classification task using logistic regression and a large pretrained language model. By including prompt, argumentative type and essay context as additional features, we show that the quality prediction performance benefits from this information.

In Study 2, we learn the span detection, type and quality prediction tasks in different settings. Comparing to a setup where the arguments were first segmented and then classified into different types and quality scales individually, we find that multi-task learning settings have a better performance.

Our source-code is publicly available at https://github.com/yuningDING/multi-task-argument-mining.

2 Related Work

With the annotations in the PERSUADE Corpus, argument scoring can be seen as a three-step process: the arguments spans are first detected, then classified into different argumentative types and lastly assessed with different quality labels. We structure related work accordingly and begin with work on argument mining, which concerns the first two steps together, before moving to argument quality prediction. Finally, we look at the application of multi-task learning in the educational domain.

2.1 Argument Mining

Toulmin's argumentation model (Toulmin, 1958, 2003) and variations of it are a main theoretic framework used for argument mining in various domains. One of the first approaches focusing on automatically detecting arguments in student essays is the one by Stab and Gurevych (2014a), who simplified the model to three labels (*premise, claim* and *major claim*) and used it to annotate a corpus of 90 essays. Their span classification approach using a Support Vector Machine (SVM) reached an F1-score of .72 (Stab and Gurevych, 2014b). Persing and Ng (2015) followed this schema to annotate the International Corpus of Learner English (Granger et al., 2009) and trained classifiers to predict the argument boundaries with an average F1-score of .57 (Persing and Ng, 2016).

The PERSUADE corpus (Crossley et al., 2022) used in this paper uses yet another version of the Toulmin model (Nussbaum et al., 2005; Stapleton and Wu, 2015) with seven argumentative elements, namely *lead, position, claim, counterclaim, rebuttal, evidence* and *concluding statement*. In order to detect these labels, Ding et al. (2022) trained sequence tagging models using pretrained Longformer (Beltagy et al., 2020) on different subsets of this corpus and reported a F1-score of .55 Their framework will be utilized in our experiments.

2.2 Argument Quality Prediction

For argument quality prediction, we first introduce quality criteria before looking into applications in the educational domain. We disregard the huge body of work using argumentation features to predict the overall score of an essay (e.g. Ghosh et al., 2016; Persing and Ng, 2016; Nguyen and Litman, 2018), but focus on approaches assessing argument quality specifically.

Although there is no common definition of argument quality, criteria proposed in the literature can be classified into two groups (Wachsmuth et al., 2017). Those measuring the *logical quality* of arguments, such as relevance, acceptability, sufficiency (Johnson and Blair, 2006) or structural well-formedness (Damer, 2012) and those measuring the *rhetorical quality*, which is operationalized by criteria such as effectiveness (Blair, 2011). Both of these two types of quality have been investigated in student essays.

In terms of logical quality, Ong et al. (2014) generate a rule-based score for the *completeness* of argumentation by rewarding an essay for occurrences of different argument types. Rahimi et al. (2014) introduce an *evidence* score, assessing how well the essay's stance is supported by the given facts on a scale from 1-4. They predict this score with a Random Forest model, reaching a Quadratic Weighted Kappa (QWK) score of .64. Stab and Gurevych (2017) annotate arguments in student essays as *sufficient* or *insufficient* and report convolutional neural network results with a Macro F1-score of .83.

In terms of the rhetorical quality, Persing and Ng (2015) evaluate argument *strength* with a numerical score from one to four using an SVM. Similar approaches were applied on thesis *clarity* and argument *persuasiveness* (Persing and Ng, 2013, 2017).

Using annotations of the PERSUADE corpus, in this paper, we focus on *effectiveness*, which we will henceforth also call *quality*. Unlike the work above, we take the argument quality prediction either as a span classification task, or as a sequence tagging task in a multi-task learning setting.

2.3 Educational Multi-Task Learning

The common interpretation of multi-task learning (MTL), which we also follow in our study, comprises approaches which learn different tasks on the same data set with a combined loss. For example, a multi-task Bi-directional Long Short-Term Memory Network (BiLSTM) (Rei, 2017) was trained for the joint tasks of grammatical error detection and automated essay scoring (Cummins and Rei, 2018). Another variant of hierarchical BiLSTMs was trained for discourse element identification and the organization evaluation in Chinese student argumentative essays (Song et al., 2020). Muangkammuen and Fukumoto (2020) combined word and sentence level BiLSTMs into a hierarchical model to predict essay scores along with sentiment classes of individual words. Similar to this strategy, we utilize different annotations on the same essays and define **argument span detection, type and quality prediction** as our joint tasks.

3 Data

As mentioned above, the data in this paper stems from the Kaggle competition "Feedback Prize - Predicting Effective Arguments", which is a subset of the PERSUADE Corpus (Crossley et al., 2022). In the competition, the detailed annotation scheme is shared along with an overall inter-annotator agreement (IAA) of .73 on argument types. Although the IAA on argument quality labels remains unclear, we manually examined 10% of the essays and were convinced of the validity of the annotation on their effectiveness.

We split the data through random sampling. The splitting result is shown in Table 1. We notice that both the argument type and the quality labels are unevenly distributed: *claim* and *evidence* occur more frequently than the other types, with *counterclaims* and *rebuttals* being particularly rare. Meanwhile, *adequate* is the majority quality class. The essays in the data set have been written in response to a number of different writing prompts. We adopt the 15 individual writing prompts detected by Ding et al. (2022), who used a topic modeling approach (Angelov, 2020) and a K-means clustering (Lloyd, 1982) on tf-idf vectors (Ramos et al., 2003).

3.1 Analysis - Label Distribution

To gauge why argument quality might benefit from joint training with argument type identification, we look at the distributions of quality labels for the different argument types in the training data. Results are shown in Figure 2. While all types are most likely to be *adequate*, we see some differences between them. It is for example especially likely for positions to be adequate (68% of all positions are adequate, while the average for all other labels is

Label	#Arguments Train	Valid.	Test
Claim	9588	1165	1224
Concl. Statem.	2677	329	345
Counterclaim	1425	172	176
Evidence	9702	1187	1216
Lead	1835	235	221
Position	3210	405	409
Rebuttal	1003	121	120
Ineffective	5181	654	627
Adequate	16705	2106	2166
Effective	7554	854	918

Table 1: Data split per type and quality label.

57%). Moreover, *evidence* is the label that is most likely to be *ineffective* (26% of all evidence spans are *ineffective*, while on average 14% of the other labels are). These priors that can be derived from the training data may be informative during joint training of both tasks, a token that is likely to be of type *position* thus also increasing in likelihood of being *effective*.

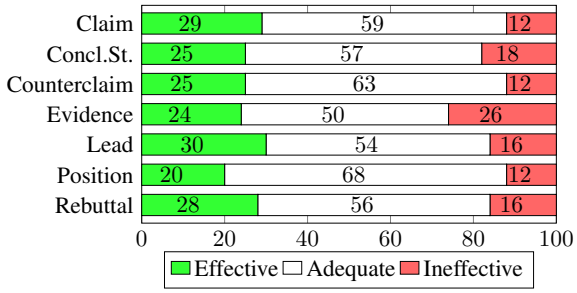

Figure 2: Distribution of the quality labels

3.2 Analysis - Structural and Semantic Similarity

To identify properties of effective or ineffective arguments, we ask whether arguments with the same quality label are similar to each other in content or structure. To answer this question, we compare pairs of gold-standard arguments according to either their structural or semantic similarity. To operationalize structural similarity, we replace every content word in an argument by their POS tag so that only function words are left intact, and then compute trigram overlap between argument pairs. For semantic overlap, we compute cosine similarity between S-BERT vectors generated using the *All-miniLM-L6-v2* model (Reimers and Gurevych, 2019). Table 2 shows both types of similarity averaged over all pairs either consisting of two *effective* arguments, two *ineffective* arguments or one *effective* and one *ineffective* argument. We only compare arguments of the same type and report results per type.

Label	Overlap ⊕⊕	⊖⊖	⊕⊖	S-BERT ⊕⊕	⊖⊖	⊕⊖
Claim	.013	.008	.010	.178	.126	.123
Concl. St.	.036	.017	.025	.213	.135	.130
Countercl.	.020	.010	.012	.219	.166	.156
Evidence	.044	.014	.025	.172	.106	.100
Lead	.024	.012	.016	.171	.117	.121
Position	.019	.011	.013	.210	.145	.141
Rebuttal	.043	.025	.032	.193	.154	.133
Average	.028	.014	.019	.194	.135	.129

Table 2: Structural (Overlap) and semantic (S-BERT) similarity of argument pairs for each argument type, averaged over all prompts. ⊖ and ⊕ stand for *ineffective* and *effective*.

In general, *effective* arguments are more similar to each other than *ineffective* ones, probably because there are more ways to "get an argument wrong" than to make it *effective*. While the absolute values are of course not comparable between structural and semantic similarity, we see that we have the highest structural similarity values for *effective* arguments of type *evidence* and *rebuttal*, while semantic similarity is highest for *position*, *concluding statement* and *counterclaim*. In general, our results suggest that *effective* arguments are easier to predict than *ineffective* ones, as they form a more coherent group.

4 Study 1 - Additional Information in Argument Quality Prediction

In a first set of experiments, we look at the task of quality prediction in isolation. In other words, we treat the task as a classification task and use gold standard information about span boundaries and - in some conditions - type of the argumentative spans. The objective of this task is to label each span with one out of three labels (*effective*, *adequate*, *ineffective*).

In doing so, we want to investigate the influence of three factors, which we assume the classification will benefit from: context, argument type and prompt information.

First, we investigate whether argument quality depends on the textual material of the respective argument alone or whether the context of the whole essay is important. We assume that the influence of

the essay context might be twofold: First, an argument could make more or less sense depending on its surrounding material. Second, we observe that 87% of all essays contain only *effective* and *adequate* or only *ineffective* and *adequate* arguments. Thus, the quality of surrounding elements could help to estimate the quality of an argument.

Second, we hypothesize that the prediction of argument quality depends on the argument type. A certain argumentative unit could be *adequate* as *lead*, but *inadequate* as *conclusion*. Furthermore, spans of different argument types have different priors to be of a certain quality level. Thus, knowing the argument type might help.

Third, we investigate whether the argument quality prediction is prompt-dependent. The performance of predictions could benefit from prompt information, because students may find certain prompts are more difficult to argument effectively than other prompts.

4.1 Experimental Setup

Classifier and features We compare shallow and deep learning methods. For shallow learning we used **logistic regression** from scikit-learn (Pedregosa et al., 2011) using 10,000 tf-idf weighted uni- to trigrams and 10-fold cross-validation. For deep learning, we fine-tuned a **pretrained BERT-model**[2]. We trained with a batch size of 8 for a maximum of 10 epochs with an Adam optimizer (Kingma and Ba, 2014) and cross entropy (BCEWithLogitsLoss) as loss function, although the optimum on the validation data was usually reached after one epoch already. For this experiment, all models together were trained in under 10 hours on a single GPU.

When integrating the three types of additional information, we proceed in the following way: For shallow learning, we appended an one-hot encoding feature representation for argument type and prompt to the n-gram feature vector. For context information, we vectorized the whole essay in the same way as the argumentative unit to be classified and append the resulting vector. For deep learning, we also appended one-hot encoding feature representations for argument type and prompt to the BERT output vector. For context information, we appended the complete essay to the input text after a separator token.

[2] https://huggingface.co/bert-base-uncased

	Log. Regr.		BERT	
	Acc.	QWK	Acc.	QWK
Baseline	.65	.41	.67	.42
+ arg. type	.66	.44	.68	.42
+ prompt	.66	.43	**.69**	.46
+ essay	.66	.47	.68	.47
+ all three	**.67**	**.50**	.67	**.49**

Table 3: Influence of adding argument type, prompt and context information.

Evaluation We evaluate the classification using accuracy and QWK (where we treat the three argument quality labels as *ineffective*=0, *adequate*=1 and *effective*=2 points).

4.2 Results

Table 3 shows the averaged results of 10 folds in **logistic regression** or 10 epochs in **BERT**, when the three pieces of additional information are added separately or all at once. Adding each of the additional features individually already improves performance, adding all of them together leads to the largest improvement for both classifiers.[3]

5 Study 2 - Multi-Task Learning

In Study 1, we treat quality prediction as a classification task using gold-standard spans as input. Since the type information was annotated on the same spans, we also trained a classification model using the same architecture in the +**essay** setting to predict their type. Results of these two classification experiments (**Gold-standard Span** $\rightarrow$ **Type**, **Gold-standard Span** $\rightarrow$ **Quality**) are the upper bound of type and quality prediction, because such gold-standard spans are usually not available in a realistic classroom setting. Instead, spans and their type and quality need to be predicted on raw text. Aiming to explore different methods to learn these labels, either jointly or sequentially, we designed the following study.

5.1 Experimental Setup

In the **Span** $\rightarrow$ **Type** and **Span** $\rightarrow$ **Quality** baseline setups, we first detect spans in a **sequence tagging** step and then classify them as different types and quality levels separately. The right arrow $\rightarrow$ indicates that we take the detected spans from the first

[3] Along with the experiments above, we trained classifiers for each prompt and each argument type separately. However, most of these classifiers only predicted the majority class due to limited amount of training data, so we do not present the detailed results here.

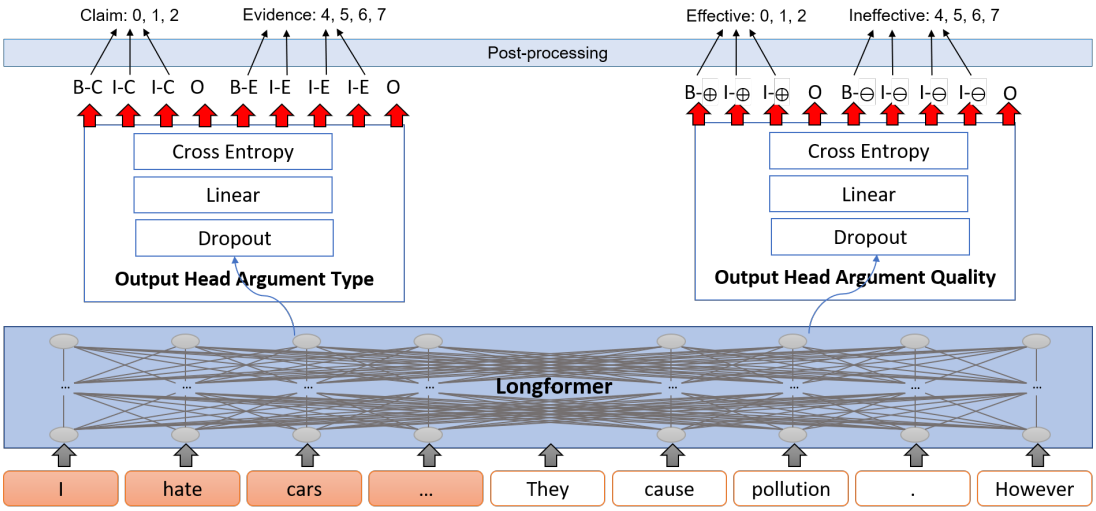

Figure 3: Architecture of multi-task sequence tagging.

step as the input for the following **classification** step(s).

Next, we combine the two classification tasks from the baseline into a multi-task learning problem (**Span** $\rightarrow$ **(Type, Quality)**$_{MTL}$). We also merge either one of the classification tasks with the sequence tagging: **Span+Type** $\rightarrow$ **Quality** detects argument spans of different types before predicting their effectiveness, while **Span+Quality** $\rightarrow$ **Type** predicts argument types after detecting effective or ineffective spans. Finally, in the **(Span+Type, Span+Quality)**$_{MTL}$ setup, span, type and quality are learned jointly in a multi-task sequence tagging approach.

Architectures For **sequence tagging**, we use a longformer-based architecture developed by Ding et al. (2022) predicting an Inside-Outside-Beginning (IOB) tag for each token. Tokens were classified into $2n + 1$ classes for a prediction task in n classes, because we have a B-class and I-class for each label, as well as an additional class (called O class) for tokens that do not belong to one of the classes. This architecture is modified for multi-task learning in the **(Span+Type, Span+Quality)**$_{MTL}$ setup by adding one additional classification head with equal weight in the loss computation after sharing the longformer block between both tasks, as shown in Figure 3. In order to evaluate the predictions on the span level, this architecture includes a post-processing step, which assembles tokens into segments predictions with token indices.

For **classification**, we use the architecture in the + **essay** setting from Study 1. Similarly to the multi-task approach for sequence tagging, we add a second classification head to the BERT model, in order to classify argumentative type and quality labels at the same time.

In all setups, we train for 10 epochs for each task, evaluating after every epoch on the evaluation data and choosing the model that maximizes the F1-score, or in the case of multi-tasking, the average F1-score of both tasks. We report results from averaging over three individual runs to avoid randomization artifacts. The total training and inference time were around 110 hours on a single GPU.

Evaluation As we treat one or both of the classification tasks as sequence tagging in some settings, we must evaluate them differently from Study 1. We follow the evaluation method proposed by the Kaggle competition: A predicted span with at least 50% overlap with a gold span of the same type is considered a true positive, while unmatched gold spans are counted as false negatives and unmatched predictions as false positives. The final score is computed as the macro-averaged F1-score across all seven argument types for type prediction and across all three quality labels for quality prediction. As the F1-measure does not reward a higher overlap once the predicted span matches at least 50% of the gold span, we also evaluate accuracy on the token level.

5.2 Results

Table 4 presents the results of this study.

Without the gold-standard spans, span labeling in the baseline setting reaches a token accuracy of .95, because most of the tokens can be easily pred-

Setting		Span		Type		Quality	
		F1	Acc	F1	Acc	F1	Acc
Gold-standard Span → Type		1	1	.71	.82	-	-
Gold-standard Span → Quality		1	1	-	-	.51	.69
Baseline: Span → Type		.63	.95	.37	.62	-	-
Baseline: Span → Quality		.63	.95	-	-	.33	.57
Span → (Type, Quality)$_{MTL}$		**.63**	**.95**	.33	.60	**.39**	**.56**
Span+Type → Quality		.62	.89	**.53**	**.73**	.34	.58
Span+Quality → Type		.58	.84	.39	.59	.35	.63
(Span+Type, Span+Quality)$_{MTL}$	Type	**.89**	**.95**	**.53**	**.73**	**.39**	**.65**
	Quality	.87	.95				

Table 4: Performance on span detection, type and quality prediction in different setups.

icated correctly as the majority class ("I-Span"). However, the boundaries between spans cannot be precisely positioned, leading to a F1-score of .63. Based on these predicted spans, the results of the argumentative type and quality classification are not optimal.

By jointly learning the two classification tasks (**Span → (Type, Quality)**$_{MTL}$), the F1-score of quality prediction was improved from .33 to .39. It confirms the result in Study 1 that adding type information can benefit quality prediction. However, the type prediction didn't get any improvement through learning quality labels.

By combining the type classification task with span identification into one sequence tagging step, i.e., in the setting **Span+Type → Quality**, we see that both of the type and quality prediction performance was substantially improved compared to the baseline. In contrast, the **Span+Quality → Type** setting has the worst performance on span identification among all settings. One possible reason could be that spans and their types are more related than spans and their effectiveness.

Lastly, the result of **(Span+Type, Span+Quality)**$_{MTL}$ shows that our multi-task approach combing all three tasks provides the best performance. With the additional type and quality information, span prediction reaches its best performance. The span boundaries learned together with type information are slightly more precise than spans learned together with quality information (.89 vs .87), which also indicates a closer relation between span and type than between span and quality.

5.3 Analysis

When looking at the individual classification results of **(Span+Type, Span+Quality)**$_{MTL}$ in Table 5, we find that, in line with the results of Ding et al. (2022), arguments are overlooked rather than confused. Among the type of confusions that do exist, mixing up *claim* and *evidence* or *claim* and *position* is the most common.

Whenever an argument (other than *none*) is identified, the labels *effective* and *ineffective* are rarely confused (altogether 8 times), much more confusion arises from neighboring quality labels. The following examples illustrate that these labels are in fact often hard to distinguish. Both are correctly recognized as *positions*, but mislabeled regarding their quality:

(3) *I think that we should change the voting system from the Electoral college to the most popular vote.*
(Gold: *effective*, predicted: *adequate*)

(4) *I believe that we, the people, should have the opportunity to choose whom we wish to become President.*
(Gold: *adequate*, predicted: *effective*)

While both positions make the same point of wanting to adopt the system of using a popular vote, the first one also mentions the current way of using the Electoral College, whereas the second one expresses the same position in a more indirect, albeit more passionate, way.

If an argument is not found (i.e. the predicted type is *none*), argument quality is not always *none* and disproportionately often wrong when compared to predictions of argument types other than

Pred \ Gold		Claim			Concl. S.			Countercl.			Evidence			Lead			Position			Rebuttal		
		⊖	⊙	⊕	⊖	⊙	⊕	⊖	⊙	⊕	⊖	⊙	⊕	⊖	⊙	⊕	⊖	⊙	⊕	⊖	⊙	⊕
Claim	⊖	6	6	-	-	-	-	1	-	-	1	1	-	-	-	-	1	1	-	-	-	-
	⊙	25	122	18	-	2	1	1	5	-	10	12	1	-	3	-	3	8	2	-	-	-
	⊕	-	23	66	-	-	-	-	1	-	1	1	3	-	-	-	-	-	1	-	-	-
	⊘	9	53	29	-	1	-	-	1	-	2	4	1	-	-	-	-	3	1	-	-	-
Conclud. Statem.	⊖	-	-	-	12	9	-	-	-	-	2	-	-	-	-	-	1	1	-	-	-	-
	⊙	-	1	-	9	107	12	-	1	-	2	3	-	-	-	-	1	1	-	1	-	-
	⊕	-	-	-	1	16	44	-	-	-	-	-	1	-	-	-	-	-	-	-	-	-
	⊘	-	-	-	3	20	12	-	1	-	1	1	-	-	-	-	-	1	-	-	-	-
Counter claim	⊖	-	1	-	-	-	-	1	2	-	-	-	-	-	-	-	-	-	-	-	-	-
	⊙	2	7	-	-	-	-	4	2	28	2	2	-	-	-	-	-	1	-	-	-	-
	⊕	-	-	1	-	-	-	-	8	14	-	-	-	-	-	-	-	-	-	-	-	-
	⊘	1	2	1	-	-	-	2	10	3	-	1	-	-	-	-	-	-	-	-	-	-
Evidence	⊖	3	3	-	1	3	-	-	2	-	65	35	1	1	1	-	-	-	-	2	1	-
	⊙	5	20	2	1	3	-	-	3	1	46	192	24	-	1	-	1	3	1	-	5	-
	⊕	-	1	4	-	-	1	-	-	-	3	51	159	-	-	1	-	-	-	-	1	2
	⊘	3	10	3	1	3	1	-	3	1	28	60	25	1	1	-	-	1	-	-	4	1
Lead	⊖	-	-	-	-	-	-	-	-	-	-	-	-	3	3	-	-	-	-	-	-	-
	⊙	1	-	1	-	-	-	-	-	-	2	2	-	9	61	14	2	5	-	-	-	-
	⊕	-	-	-	-	-	-	-	-	-	-	1	-	-	11	44	-	-	1	-	-	-
	⊘	-	1	-	-	-	-	-	-	-	1	1	-	1	7	7	-	1	-	-	-	-
Position	⊖	-	-	-	-	-	-	-	-	-	-	-	-	1	-	-	2	2	1	-	-	-
	⊙	1	9	1	-	2	-	-	-	-	1	1	-	2	4	-	7	104	16	-	-	-
	⊕	-	4	1	-	-	-	-	-	-	-	-	-	-	-	-	-	8	17	-	-	-
	⊘	-	3	1	1	-	-	-	-	1	-	1	-	2	1	-	2	22	9	-	-	-
Rebuttal	⊖	-	-	-	-	-	-	-	-	-	-	1	-	-	-	-	-	-	-	2	4	1
	⊙	-	1	-	-	-	-	-	-	-	1	1	-	-	-	-	-	-	-	1	9	2
	⊕	-	-	1	-	-	-	-	-	-	-	1	-	-	-	-	-	-	-	-	4	7
	⊘	6	26	3	1	4	-	1	5	-	11	28	1	1	-	-	-	7	-	2	4	2
none	⊖	32	131	61	7	9	4	2	17	3	45	71	11	5	4	1	7	37	9	4	11	3
	⊙	11	53	20	1	8	2	1	8	2	14	53	12	2	5	1	1	9	3	-	4	3
	⊕	1	11	15	1	2	2	-	1	1	10	14	7	-	1	1	-	1	4	1	1	2
	⊘	102	473	229	27	86	37	9	71	17	153	313	120	24	45	24	27	165	42	11	47	13

Table 5: Confusion matrix for multi-task learning of argument span, type and quality, with ⊖, ⊙, ⊕ and ⊘ denoting *ineffective*, *adequate*, *effective* and *none* quality prediction, respectively. Cells highlighted in green indicate that both type and quality were predicted correctly, while blue (purple) cells show where only the argument type (quality) classification is correct.

none, indicating a general difficulty to handle these spans regarding both tasks. In case an incorrect argument type other than *none* is predicted, the quality label is still correct about half of the time.

Setting	Label	P	R	F
MTL	Adequate	.43	.49	.46
	Effective	.40	.51	.45
	Ineffective	.26	.28	.27
	overall	.40	.45	.39
Span + Quality	Adequate	.46	.44	.45
	Effective	.33	.45	.38
	Ineffective	.20	.27	.23
	overall	.38	.42	.35

Table 6: Precision, Recall and F1-score in the MTL (both) and the MTL (quality only) conditions.

In an effort to understand how learning quality labels benefits from argument type information in the multi-task learning setting, we inspect how the quality classification results are influenced by the inclusion of the other task. Therefore, we in Table 6 compare the multi-task and span+quality results in more detail. When comparing the performance of the individual quality, all three labels show improvement in their F1-scores, with the majority label *adequate* showing the smallest increase of .01, while the inceases are larger for *ineffective* (.04) and *effective* (.07) arguments. Precision of classifying *effective* and *adequate* arguments improves more (.06 and .05) than it does for *ineffective ones* (.01). Interestingly, jointly learning argument types does not increase false positives for *ineffective* and *effective* arguments, but it does for *adequate* ones. Still, this is evened out by an increase in true positives, which is pronounced enough to overall lead to the observed in increase in F1-score. On a side note, our findings in the dataset analysis, namely that *effective* arguments are more similar to each other than *ineffective* ones, is confirmed here by a higher F1-score of *effective* arguments compared to *ineffective* ones.

When evaluating the token classification results in a binary fashion, only distinguishing between B- and non-B tokens, recall and precision increase by .01 and .04, respectively. The joint training is thus

leading to a more precise recognition of argument borders between quality spans, perhaps due to certain argument types being more tightly associated with certain keywords (Ding et al., 2022).

6 Conclusion

In this paper, three tasks in automatic scoring of argumentative essays were examined, namely argument span detection, type and quality prediction. We found that the quality prediction benefits from prompt information as well as essay context. We further found that, compared to a setup where the arguments were first segmented and then classified into different types and quality scales, multi-task learning settings performed better.

Limitations

A fraction of 157 essays was too long to fit into our transformer model so that arguments later in the text have not been identified at all.

As gold standard information about the writing prompt for a specific essay was not released with the dataset, we had to rely on automatically assigned prompt information with an estimated average accuracy of 0.97 according to Ding et al. (2022). In a realistic class-room setting, however, the information about the writing prompt would be readily available, thus we probably underestimated the effect of adding prompt information.

We tested our models on the PERSUADE dataset of English high-school writings only, thus we cannot be sure whether results transfer to other educational contexts and languages. We will address this further in future work, where we aim at using essays in German and from EFL contexts as well.

Since our model was trained on a limited amount of data, it may have the potential risk of discouraging students to write innovative, but effective arguments. As discussed in automatic essay scoring approaches, computers may be able to analyze writing for the presence or absence of certain words or structures (in our case arguments), but they cannot really understand or appreciate a writer's message in the same sense that human readers can (Powers et al., 2002).

Acknowledgements

This work was partially conducted at "CATALPA - Center of Advanced Technology for Assisted Learning and Predictive Analytics" of the FernUniversität in Hagen, Germany, and partially within the KI-Starter project "Explaining AI Predictions of Semantic Relationships" funded by the Ministry of Culture and Science, Nordrhein-Westfalen, Germany.

References

Richard Andrews. 2009. *Argumentation in higher education: Improving practice through theory and research*. Routledge.

Dimo Angelov. 2020. Top2Vec: Distributed Representations of Topics. *arXiv e-prints*, pages arXiv–2008.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

J Anthony Blair. 2011. *Groundwork in the theory of argumentation: Selected papers of J. Anthony Blair*, volume 21. Springer Science & Business Media.

Scott A Crossley, Perpetual Baffour, Yu Tian, Aigner Picou, Meg Benner, and Ulrich Boser. 2022. The persuasive essays for rating, selecting, and understanding argumentative and discourse elements (persuade) corpus 1.0. *Assessing Writing*, 54:100667.

Ronan Cummins and Marek Rei. 2018. Neural multitask learning in automated assessment. *arXiv preprint arXiv:1801.06830*.

T Edward Damer. 2012. *Attacking faulty reasoning*. Cengage Learning.

Yuning Ding, Marie Bexte, and Andrea Horbach. 2022. Don't drop the topic - the role of the prompt in argument identification in student writing. In *Proceedings of the 17th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2022)*, pages 124–133, Seattle, Washington. Association for Computational Linguistics.

Debanjan Ghosh, Aquila Khanam, Yubo Han, and Smaranda Muresan. 2016. Coarse-grained argumentation features for scoring persuasive essays. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 549–554.

Sylviane Granger, Estelle Dagneaux, Fanny Meunier, Magali Paquot, et al. 2009. *International corpus of learner English*, volume 2. Presses universitaires de Louvain Louvain-la-Neuve.

Ralph Henry Johnson and J Anthony Blair. 2006. *Logical self-defense*. Idea.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Stuart Lloyd. 1982. Least squares quantization in PCM. *IEEE transactions on information theory*, 28(2):129–137.

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.

Huy Nguyen and Diane Litman. 2018. Argument mining for improving the automated scoring of persuasive essays. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

E Michael Nussbaum, CarolAnne M Kardash, and Steve Ed Graham. 2005. The effects of goal instructions and text on the generation of counterarguments during writing. *Journal of Educational Psychology*, 97(2):157.

Nathan Ong, Diane Litman, and Alexandra Brusilovsky. 2014. Ontology-based argument mining and automatic essay scoring. In *Proceedings of the First Workshop on Argumentation Mining*, pages 24–28.

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.

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

Isaac Persing and Vincent Ng. 2015. Modeling argument strength in student essays. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 543–552.

Isaac Persing and Vincent Ng. 2016. End-to-end argumentation mining in student essays. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1384–1394.

Isaac Persing and Vincent Ng. 2017. Why can't you convince me? modeling weaknesses in unpersuasive arguments. In *IJCAI*, pages 4082–4088.

Donald E Powers, Jill C Burstein, Martin Chodorow, Mary E Fowles, and Karen Kukich. 2002. Stumping e-rater: challenging the validity of automated essay scoring. *Computers in Human Behavior*, 18(2):103–134.

Zahra Rahimi, Diane J Litman, Richard Correnti, Lindsay Clare Matsumura, Elaine Wang, and Zahid Kisa. 2014. Automatic scoring of an analytical response-to-text assessment. In *International conference on intelligent tutoring systems*, pages 601–610. Springer.

Juan Ramos et al. 2003. Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, volume 242, pages 29–48. Citeseer.

Marek Rei. 2017. Semi-supervised multitask learning for sequence labeling. *arXiv preprint arXiv:1704.07156*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Wei Song, Ziyao Song, Lizhen Liu, and Ruiji Fu. 2020. Hierarchical multi-task learning for organization evaluation of argumentative student essays. In *IJCAI*, pages 3875–3881.

Christian Stab and Iryna Gurevych. 2014a. Annotating argument components and relations in persuasive essays. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1501–1510, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Christian Stab and Iryna Gurevych. 2014b. Identifying argumentative discourse structures in persuasive essays. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 46–56.

Christian Stab and Iryna Gurevych. 2017. Recognizing insufficiently supported arguments in argumentative essays. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 980–990.

Paul Stapleton and Yanming Amy Wu. 2015. Assessing the quality of arguments in students' persuasive writing: A case study analyzing the relationship between surface structure and substance. *Journal of English for Academic Purposes*, 17:12–23.

Stephen E Toulmin. 2003. *The uses of argument*. Cambridge university press.

Stephen Edelston Toulmin. 1958. The uses of argument.

Henning Wachsmuth, Nona Naderi, Yufang Hou, Yonatan Bilu, Vinodkumar Prabhakaran, Tim Alberdingk Thijm, Graeme Hirst, and Benno Stein. 2017. Computational argumentation quality assessment in natural language. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 176–187, Valencia, Spain. Association for Computational Linguistics.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
7

☑ A2. Did you discuss any potential risks of your work?
7

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract, 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

3, 4, 5

☑ B1. Did you cite the creators of artifacts you used?
2, 3, 4, 5

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
2, 3

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
2, 3

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
2, 3

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
2, 3

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
3

C ☑ Did you run computational experiments?

4, 5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
4.1, 5.1

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
4.1, 5.1

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
4, 5

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
4.1, 5.1

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.