Using Large Language Models to Analyze Students' Collaborative Argumentation in Classroom Discussions

Nhat Tran and **Diane Litman** and **Amanda Godley**
University of Pittsburgh
Pittsburgh, PA, USA
{nlt26, dlitman, agodley}@pitt.edu

Abstract

Collaborative argumentation enables students to build disciplinary knowledge and to think in disciplinary ways. We use Large Language Models (LLMs) to improve existing methods for collaboration classification and argument identification. Results suggest that LLMs are effective for both tasks and should be considered as a strong baseline for future research.

1 Introduction

Collaborative argumentation is a key mechanism through which students engage in critical thinking and co-construct knowledge during classroom discussions (Larson, 2000; Reznitskaya and Gregory, 2013). Because well-facilitated discussions are a cornerstone of effective instruction, they are frequently a target of measurement (Matsumura et al., 2008; Hill et al., 2008; Reznitskaya and Wilkinson, 2021; Bouton and Asterhan, 2023). However, large-scale human evaluation is costly and challenging due to the complexity of collaboration and argumentation in multi-party dialogue. Thus, AI methods – across a range of measurement frameworks – are being developed to automatically assess classroom dialogue quality (Wang and Demszky, 2023; Xu et al., 2024; Kelly et al., 2018), and to develop tools for improving dialogic teaching aimed at teachers (Lugini et al., 2020; Suresh et al., 2021), coaches (Wang and Demszky, 2023), and learning scientists (Tran et al., 2024b).

The tasks of computationally analyzing students' *collaboration* and *argumentation* in a classroom discussion are challenging (Olshefski et al., 2020; Lugini and Litman, 2020; Wang and Chen, 2024; Shiota and Shimada, 2022). For our dataset (example in Table 3 and details in Section 3), collaboration analysis involves classifying every student turn as relevant to collaborative argumentation (e.g., initiating a new idea or challenging another student's claim) or not (non-argumentative). Argumentation analysis can be further divided into a pipeline of two subtasks. The first involves identifying spans of text consisting of argument discourse units (ADUs), i.e., argument component detection (ACD). The next subtask, argument component classification (ACC), focuses on assigning a label (Claim, Evidence, Warrant) to each ADU [1].

While computational argument mining is an active research area (Stede and Schneider, 2019; Lawrence and Reed, 2020), relatively little work has been done on collaborative discussions. Also, prior work often omits ACD and takes already identified argument components as input, and thus focuses on only argument component classification (ACC) rather than on end-to-end argument mining (Deguchi and Yamaguchi, 2019; Tran and Litman, 2021). Finally, argument component classification is often treated as a sequence labeling task, but it needs extensive finetuning and offers limited control over the output, especially when capturing relationships between components (Schulz et al., 2019; Alhindi and Ghosh, 2021).

To address these challenges, we leverage Large Language Models (LLMs) for two key tasks in assessing collaborative argumentation in classroom discussions. LLMs offer strong generative capabilities, enabling effective classification and sequence labeling with minimal annotated data. For *collaboration classification*, we replace traditional classifiers with LLMs and compare multi-class versus binary prompting strategies. For end-to-end argumentation identification, we use LLMs to jointly segment and classify argument components. Our study aims to answer the following questions:

RQ_1 Is LLM effective for collaboration classification?

RQ_2 Can we use LLM to perform end-to-end argument identification, and how good is it?

[1] We use ADU and argument component interchangeably.

Proceedings of the Artificial Intelligence in Measurement and Education Conference (AIME-Con) – Volume 1: Full Papers, pages 111–125
October 27-29, 2025 ©2025 National Council on Measurement in Education (NCME)

Our contributions are two-fold. First, we show that few-shot prompting enables LLMs to outperform a BERT-based collaboration classifier trained on significantly more annotated data, with binary prompting proving more effective than multi-class classification. Second, we show that LLMs can perform end-to-end argument identification, with our structure-focused evaluation highlighting their effectiveness under a simplified argument scheme (i.e., at most one Claim, Evidence, or Warrant).

2 Related Work

Much of the prior work on argument mining addressed the problems of argument segmentation (i.e., identifying ADU boundaries), component classification, and relation identification modeled in a pipeline of subtasks (Potash et al., 2017; Niculae et al., 2017). However, many of them assume the availability of segmented argumentative units and do the subsequent tasks such as classification of argumentative component types (Lugini and Litman, 2018, 2020; Garcia-Gorrostieta et al., 2018), and argument relation identification (Ghosh et al., 2016; Gemechu et al., 2024; Contalbo et al., 2024). We perform argument component segmentation and classification simultaneously by utilizing LLMs.

Previous work on argument segmentation includes approaches that model the task at a surface level by classifying sentences as argumentative or non-argumentative (Ajjour et al., 2017; Chakrabarty et al., 2019). At a more fine-grained level, there are studies that use heuristics to identify argumentative segment boundaries (Wachsmuth et al., 2016). Prior work also treats the task as a sequence labeling task by performing token-level classification to directly identify the type of the argument component and achieves promising results (Schulz et al., 2019; Alhindi and Ghosh, 2021). Additionally, multi-task learning, which utilizes other NLP tasks such as part-of-speech tagging or datasets from other domains, is a widely used tool to further boost performance of argument component classification (Daxenberger et al., 2017; Schulz et al., 2018; Mensonides et al., 2019). Unlike these approaches, we do not formulate the argument identification task as a token-level sequence labeling task. Instead, we consider it a text generation task by leveraging LLMs, which have been shown to be effective at text span extraction (Tran et al., 2024a; Wang et al., 2025).

Since LLMs such as GPT-4 (OpenAI et al., 2024), Llama (Grattafiori et al., 2024), and Mistral (Jiang et al., 2023) have outperformed pretrained language models (PLMs) such as BERT (Devlin et al., 2019) in many natural language processing (NLP) tasks, there has been growing interest in leveraging them for argument mining and text extraction. Kashefi et al. (2023) uses GPT-3 for claim and premise detection, but it is only a classification task on the sentence level. Chen et al. (2024b) explores the potential of LLMs in many argument computation tasks, but does not cover joint tasks such as end-to-end argument identification. Pichler et al. (2025) and Lin and Koedinger (2024) demonstrate that LLMs are effective in sequence labeling if they are prompted appropriately, but they do not test them in the context of argument mining. Our work leverages LLM for analyzing collaborative argumentation, focusing on 2 tasks: collaboration classification and argument identification.

3 Data

We use Discussion Tracker (DT)[2], publicly accessible classroom discussion data annotated for collaborative argumentation (Olshefski et al., 2020), for our experiments. The DT data comprises 90 transcribed multi-party discussions conducted in American high school English Language Arts classes. We use two subsets from the corpus. They were collected in 2019 (29 transcripts) and 2022 (61 transcripts) using the same annotation guidelines, so we refer to them as DT_19 and DT_22, respectively.

We use the data for two tasks: *collaboration* code classification and *argumentation* identification. Students' talk at the turn level was annotated for *collaboration*, and talk at the argument discourse unit (ADU) level was annotated for *argumentation*. Specifically, argumentative turns were annotated with one of four collaboration codes: *New Idea*, *Agreement*, *Extension*, and *Challenge*; turns that contained no substantive argumentation were labeled with the collaboration code *None*. Argumentative turns were further segmented into argument discourse units (ADUs), which were labeled for argument types: *Claim*, *Evidence*, or *Warrant*. Annotators were instructed not to segment turns into multiple claims or multiple units of evidence, and every word belongs to one ADU (i.e., no gaps between ADUs). As a result, each segmented ADU is considered an argument component.

Definitions of collaboration and argumentation

[2] https://discussiontracker.cs.pitt.edu

coding are in Tables 1 and 2. Table 3 shows an annotated transcript, while statistics are in Table 4.

4 Method

We use few-shot prompting to instruct a LLM to tackle the tasks, using the prompts in Tables 5 and 6. The few-shot examples are not from the test set; they are either from the training set (cross-validation) or from a different DT corpus (e.g., using examples from DT_19 to test on DT_22).

4.1 Collaboration Classification

The collaboration task involves classifying a student's turn into 1 of 5 classes: Non-Argumentative (None), New Idea, Agreement, Extension, Challenge. We utilize LLMs in two approaches.

LLM-multi. We treat the task as standard multi-class classification. Specifically, we ask the LLM which of the 5 classes it thinks the turn belongs to. The prompt includes the instruction, definitions of the 5 classes, and 10 few-shot examples. Each few-shot example consists of a turn and its correct class. We have 2 examples for each of the 5 classes.

LLM-binary. Although not specifically focused on collaboration classification, prior work has shown that utilizing LLM is more effective at binary classification compared to multi-class classification on classroom discussion data (Tran et al., 2024b). Thus, we perform 4 binary classification tasks for each student's turn. For an argumentative class X (New Idea, Agreement, Extension, and Challenge), we ask the LLM a yes/no question about whether the turn is considered X by providing it with X's definition. We call the set of the remaining argumentative classes except X as S. For instance, if X is New Idea, S = {Agreement, Extension, Challenge}. For few-shot examples, we provide 5 examples where a turn should be predicted as X (positive examples) and 5 examples where it should not be (negative examples). In the 5 negative examples, we use 1 example where the turn's gold-standard class is s_i for all $s_i \in S$ and 2 examples where the turn's class is None. For the final turn-level prediction, if the LLM predicts 'no' for all of the 4 argumentative classes, it is a non-argumentative turn (None). If there is more than one class predicted as 'yes', we select one with the highest probability, p(yes|X).

4.2 Argumentation Identification

This task is typically approached as a two-step pipeline applied to argumentative student turns. The first step, argument component detection (ACD), involves identifying spans of text that constitute argument discourse units (ADUs). The second step, argument component classification (ACC), assigns a label (Claim, Evidence, or Warrant) to each identified ADU. One way to solve two subtasks simultaneously is to treat them as a sequence labeling task using the BIO scheme (Beginning, Inside, or Outside) (Schulz et al., 2019; Alhindi and Ghosh, 2021). Specifically, instead of segmenting the text into ADUs first, we can conduct a token-level[3] classification task to identify the type of the argument component (e.g., B/I tokens from claim, evidence, and warrant) directly by joining the first and the second sub-tasks in a single task (i.e., B-Claim, I-Claim, B-Evidence, ...). See Figure 1 for an illustration of the BIO conversion. However, since LLMs are potent tools for following human instructions, prior work utilizing LLMs for sequence labeling employs generative approaches instead of performing the traditional token-level classification task (Lin and Koedinger, 2024; Wang et al., 2025). Also, due to the nature of the dataset, every word in an argumentative turn belongs to either Claim, Evidence, or Warrant (i.e., no O labels are present). Thus, we treat the task as a text generation task for the LLMs to perform both ACD and ACC tasks simultaneously.

LLM-auto. We let the LLM extract non-overlapping text spans of the target turn into C, E, and W. Because 95% of turns had a collaborative relationship with turns within the previous four turns (Olshefski et al., 2020), we provide four previous turns for the dialogue context, along with the definitions of C, E, and W for reference. The output is formatted as Claim: {claim_span}, Evidence: {evidence_span}, Warrant: {warrant_span}. Because an argumentative turn does not necessarily consist of all three segments (e.g., only C and E), the output text spans can be empty. We also ensure that all segmentation scenarios are covered in the few-shot examples by including at least one example for each class combination. Specifically, if we consider a scenario as a combination of C, E, and W, along with their order of appearance in the text from left to right, there are 10 scenarios in the dataset: (C), (E), (C, E), (E, C), (C, W), (E, W), (C, E, W), (C, W, E), (E, C, W), and (E, W, C). We provide one example for each of the scenarios.

LLM-refine. Previous studies show that (i)

[3] We use words as tokens.

LLM is more effective with more detailed instructions (Tran et al., 2024b) and (ii) LLM is good at judging LLM's generated answers (Chen et al., 2024a; Huang et al., 2025). We assume that LLM is better at the task when the correct combination of C, E, and W is provided. In other words, if the LLM knows that the turn only contains C and E, it provides a better segmentation than it does without that information. First, we use LLM to generate multiple argument segmentations (**LLM-gen**) given the combinations of C, E, and W (e.g., segment the text into Claim and Evidence). We ignore the ordering of C, E, and W in the combination, but instead provide different orderings in the few-shot examples. Therefore, each turn will be segmented into one of the following six combinations: (C), (E), (C, E), (C, W), (E, W), (C, E, W). Since (C) and (E) simply require marking the entire turn as C or E, we do not need LLM to do so. As a result, each turn will be segmented into four different ways by the LLM. The prompt for the first step consists of four previous turns as the dialogue context, the definitions of C, E, and W, and a specific argument combination we want to split the text into. The second step, the refinement step (**LLM-judge**), involves selecting the most suitable segmentation from the six generated options. To do so, we consult another LLM to select the best segmentation from the six options.

LLM-acc. Since prior work often only focused on the argument component classification (ACC) task (Lugini and Litman, 2020; Kashefi et al., 2023; Garcia-Gorrostieta et al., 2018; Hidayaturrahman et al., 2021) and assumed that the correct segmentation is given, we additionally conduct an experiment on using LLM specifically for argument component classification. For an ADU, we prompt the LLM to classify it as C, E, or W. Similar to other LLM approaches, we provide the 4-turn dialogue context, definitions of C, E, and W, along with 9 few-shot examples (3 of each type C, E, and W).

5 Experimental Setup

5.1 Baseline Models

Collaboration. We train a **BERT** model to predict whether a turn is either a New Idea, Agreement, Extension, Challenge, or Non-argumentative.

Argumentation. For the *argument component classification* task in which the correct argument component segmentation is provided, we compare our LLM's results (**LLM-acc**) with results from a BERT-based model utilizing local context and speaker context from Lugini and Litman (2020) (**BERT-context**). For the downstream *argument identification task* (argument segmentation + classification), we follow prior work and use BERT for sequence labeling as a baseline (Schulz et al., 2018; Kashefi et al., 2023). We call it **BERT-BIO**, which employs a BIO classification scheme to identify and classify argument components. We use BERT as the base transformer model and train a token-level classifier head on top. This baseline aims to label each token as B-Claim, I-Claim, B-Evidence, I-Evidence, B-Warrant, or I-Warrant.

We note that the BERT-context's results are from a publication using an older version of the DT_19 data (Lugini and Litman, 2020), which is no longer available. Our DT_19 version, which is corrected for better consistency, has 10 more ADUs (3145 versus 3135) compared to their version. However, because the difference is small, we still use the previously published BERT-context results to compare with our models' performance on the DT_19 data.

All BERT models are `bert-base-uncased`[4].

5.2 Experiment and Evaluation

We compare the performance of LLM and baseline approaches to answer the two research questions. mentioned in Section 1.

For collaboration prediction and argument component classification, we use the F_1 score as our evaluation metric since it is a standard multi-class classification task. We also report results in predicting Argumentative and Non-Argumentative turns.

For argument identification (segmentation + classification), due to our limited resources, we only conduct experiments on the larger corpus DT_22. After converting LLM's outputs to the word-level BIO format (see Figure 1 for an example), we can treat the task as a word-level classification task and compute the weighted F_1 score. We decided to use weighted F_1 because finding the exact boundaries of each segment is not essential empirically.

We also propose a new metric for argument identification on the component level. In a real-world application of an automated argument identification system (e.g., creating teacher dashboard analytics such as how many student claims were supported by evidence (Lugini et al., 2020)), it is more crucial to capture the structure of argument components within a single turn than to find the exact

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

splits. This is applicable to our data, as there are at most three argument components that cover every word in a turn. The word-level F_1 score does not consider component-level matching, whereas metrics like seqeval (Nakayama, 2018), which are popular for sequence labeling tasks such as named entity recognition, only consider strict matching between boundaries. We want to know whether the automated segmentation and classification have the same argument components, while not too strict in finding the boundaries between them (e.g., it is fine to have the two last words from Evidence identified as part of Warrant). To do so, we modify the metric from SemEval-2013 (Segura-Bedmar et al., 2013). Given a threshold K, a true positive (TP) is counted when the predicted span (*pred_span*) has the same label as the gold-standard span (*gold_span*) and they overlap at least $K\%$. The overlapping is calculated as $\frac{|\text{pred\_span} \cap \text{gold\_span}|}{max(|\text{pred\_span}|, |\text{gold\_span}|)}$, where $|\cdot|$ denotes the number of words in a span. Then, we can calculate Precision, Recall and F_1 normally.

The value of K controls how strictly we want the spans to match. At K = 100, we require an exact match between the two spans (i.e., same boundaries and same label) for a TP. At K = 0, we only compare predicted labels (C, E, W) with the gold-standard ones for a given turn. For example, if we predict a turn has one C and one E, as long as the gold-standard consists of exactly one C and one E, it is a correct prediction. We call this new metric Argument Component Score at K (**ACS@K**).

All experiments, including the baselines, are conducted using the same 10-fold cross-validation split provided by the DT corpus. Due to our limited resources, we utilize LLama3-8B (Grattafiori et al., 2024) as our LLM for all tasks [5].

6 Results and Discussion

6.1 Collaboration Results ($\mathbf{RQ_1}$)

Table 7 shows the macro-F_1 over 10-fold cross-validation for the collaboration prediction task on both DT_19 and DT_22. Both LLM approaches significantly outperform the BERT baseline on both datasets. Additionally, LLM-Binary is significantly better than LLM-multi in all categories ($p < 0.05$), suggesting that using multiple LLMs as binary classifiers is an effective approach (Tran et al., 2024b). On the other hand, LLM approaches are not significantly better than BERT in classifying Argumentative and Non-Argumentative turns (except for LLM-binary in DT_22). It implies that the BERT model is not inferior in identifying argumentative turns, but struggles to predict the correct labels among the four collaboration codes. Using Cohen's kappa as the metric (Table 8), we get similar observations as the two LLM approaches constantly outperform BERT, and LLM-binary consistently achieves the best performance.

Looking into Table 9, the higher weighted F_1 scores compared to macro F_1 (Table 7) indicate that the models perform better on more frequent classes. We observe that the LLM approaches significantly outperform BERT in New Idea, Extension, and Challenge. Among these three classes, New Idea and Challenge are consistently the bottom 2 for all models. We also witness opposite cases for the two minority classes that take up less than 10 % of the data on both datasets, Agreement and Challenge. For Agreement, while LLM-binary is superior compared to BERT, BERT is not significantly worse than LLM-multi, and it even surpasses LLM-multi on DT_22. We hypothesize there are lexical clues (e.g., "I agree ...") for Agreement, and the increase in training data for Agreement in DT_22 (177 versus 38 instances) helps BERT learn to recognize the pattern of this type of collaboration. On the other hand, both BERT and LLM approaches struggle with Challenge, suggesting that the difficulty does not come from the scarcity of the class (i.e., LLM models need no training data). For Extension, while LLM-multi and LLM-binary's results suggest that it is easier than Agreement, BERT finds the opposite, and the largest performance gap between BERT and LLM approaches also falls in this category. This implies that BERT is not as effective in distilling knowledge to identify Extension after training as an LLM with few-shot prompting.

6.2 Argumentation Results ($\mathbf{RQ_2}$)

For Argument Component Classification (ACC) on DT_19 [6], BERT-context (Lugini and Litman, 2020) and LLM-acc achieve 77.4 and 80.2 macro-F_1 scores, respectively. This suggests that LLM is not particularly better than BERT in classifying C, E, and W when the correct ADUs are provided.

However, when we have to perform the full Argument Identification task from scratch, which includes ACD (segmentation) and ACC (classification), we observe some performance gaps between LLM and the BERT-BIO baseline. In terms of

[5] https://huggingface.co/meta-llama/Llama-3.1-8B

[6] The two DT_19 corpora are slightly different as mentioned at the end of Section 5.1.

token-level F_1 score (Table 10), both LLM approaches outperform BERT-BIO, suggesting that utilizing the generative capability of LLM has advantages over sequence labeling with a transformer like BERT. On one hand, it is not feasible to control the tagging process of BERT-BIO at inference time. As a result, there are cases in which it provides more than one Claim, Evidence, or Warrant for a turn, which violates the nature of the DT corpus used for testing. On the other hand, we can restrict the output of LLM approaches by giving the instructions in the prompts and few-shot examples, which prevents them from violating the aforementioned data constraint. Thus, this can be one reason for the inferior performance of BERT-BIO in terms of word-level F_1 scores.

The score of the Beginning of a segment (B-C/E/W) is always lower than the Inside counterparts (I-C/E/W), which implies that it is hard to find the exact segmentation boundaries. However, B-E has higher results compared to B-C and B-W, demonstrating that the models are more effective at finding the beginning of Evidence. We hypothesize that certain words (e.g., 'because') can signal the start of evidence, making it easier to detect when students begin providing it. Among the C, E, and W, W appears to be the most challenging class to correctly identify, as the results of B-W and I-W are lower than those of the other two. Furthermore, LLM-refine significantly outperforms LLM-auto in average weighted F_1 ($p = 0.03$), suggesting that LLM is good at judging argument identification.

Figure 2 presents the proposed metric ACS@K with various K. Similar to the average weighted F_1 score (Table 10), LLM-refine beats LLM-auto and BERT-BIO. While the results of LLM-auto and LLM-refine are quite close, the BERT-BIO baseline yields noticeably lower performance. The discrepancies between BERT-BIO and the two LLM models are also larger compared to Table 10. In other words, LLM approaches are even more effective when evaluated on the argument component level. When the argument is simplified (i.e., only one C, E, and W), lacking control over the output by treating the task as a sequence labeling task (BERT-BIO) makes the argument identification results less desirable. In addition, LLM approaches are more robust when the threshold K is varied. We observe most increases in ACS@K score for LLM approaches until about $K = 40$, after which the curve remains more stable. Based on that observation, we hypothesize that LLMs might not be effective at finding exact segmentations, but are good at identifying argument components in the correct order. For example, assume the gold-standard labels for the turn from left to right are C, E, and W. If the model predicts a different order (e.g., E, C, W), it is considered correct when $K = 0$. As we increase K, that answer becomes incorrect because the overlaps between text spans do not satisfy the increased threshold. However, the graph shows that there are no big differences between $K = 40$ and $K = 0$ for the LLM approaches. This implies that the models get the argument components in the correct order. Lowering K after 60 does not show noticeably higher ACS@K scores, which further implies that the predicted argument components already have good overlap with the gold standard.

7 Conclusion

In this work, we experimented with LLMs in two classroom discussion assessment tasks: turn-level collaboration classification and end-to-end argument identification. The results show that LLMs outperform the BERT baselines in both tasks. For collaboration classification, we observe that different ways of formulating the task (binary versus multi-class classification) have an impact on performance, as the former yields better results. For argument identification, instead of dividing the task into two individual subtasks of ACD and ACC, we utilize LLMs to perform text generation to solve them simultaneously and achieve promising results.

Our results show that LLMs are robust under ACS@K, indicating they capture the correct order of argument components. Instruction following further allows finer control over argument constraints in LLMs, unlike sequence labeling with models like BERT. Future work includes fine-tuning LLMs, exploring diverse prompting strategies (e.g., Chain-of-Thought (Wei et al., 2022), example-retrieval (Wang et al., 2024), zero-shot methods), and applying these assessments downstream.

Acknowledgments

This material is based upon work supported by the National Science Foundation (NSF) under Grant # 1917673. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF. The authors would like to thank Yang Zhong and the anonymous reviewers for their valuable feedback on this work.

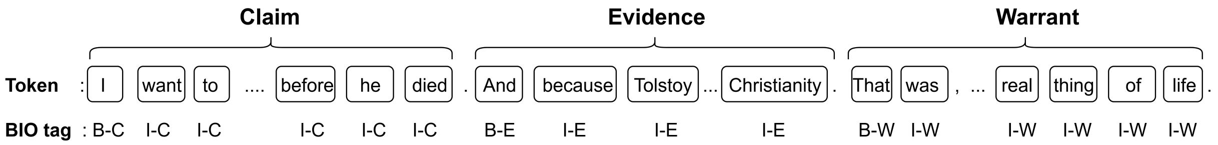
Claim
Evidence
Warrant
Token : I want to .... before he died . And because Tolstoy ... Christianity . That was , ... real thing of life .
BIO tag : B-C I-C I-C I-C I-C I-C B-E I-E I-E I-E B-W I-W I-W I-W I-W I-W

Figure 1: Conversion to BIO format. Each token is tagged as X-Y, where X is either B (Beginning) or I (Inside), and Y is either C (Claim), E (Evidence), or W (Warrant).

Code	Definition
New Idea	An initiating turn is the expression of a new idea in the discussion. This does not have to be a new topic, but should be a new idea, concept, or perspective. It usually does not reference ideas in prior turns at talk, or it does so only superficially. Turns that build on ideas in previous turns at talk are coded as "Extension". New student questions posed to the whole class that do not probe or question a previous answer are uncoded.
Extension	A turn is an extension if it builds off another student's ideas. Extension turns must extend one of the preceding four codeable student turns unless a turn prior to those 4 is specifically referenced. Extension turns include at least 2 key ideas or terms that were voiced by another student. Key ideas/terms may be textual, topical or conceptual terms. Textual terms may include characters and places from a text under discussion (like "Macbeth" or "Birnam Wood"), but do not include titles of texts. Topical terms may include disciplinary topics (like theme, metaphor, symbol, etc.). Conceptual terms may include abstract ideas (like "culture," "domination," "regret"). Extensions sometimes (but not always) include terms like "also, another, too"; or indicators of agreement/alignment (such as, "like X said…") Extensions can also include a self extension which is a turn of talk that adds information to or re-words one's own idea that was shared without acknowledging the idea of other speakers in close proximity.
Challenge	Challenge turns challenge or question a prior idea. Challenges should reference another student's turn in the preceding four codeable student talk turns. Challenges to points made further back are considered "New Ideas". A turn is considered a challenge if it includes both (1) keywords/concepts from previous turns (such as "culture," "domination," or "regretful") and (2) some indication of disagreement. Note that indications of disagreement can be very subtle (such as "still" or "actually" or "he did tell his sister") or more explicit (such as "I disagree", "No," "but," "however," "though"). A turn is considered a challenge if it challenges or requests more information, detail, elaboration, or clarification/explanation in the form of a question ("Why do you think that?" "You really think Macbeth wasn't crazy?" or "What do you mean?"). Will often include second person pronoun or direct address. Does not include procedural questions like "Wait what was his question?". Turns sometimes contain what may appear to be indications of disagreement (e.g., "however" "isn't") but are actually referring to ideas within the turn—these would likely fall under the category of "Extension".
Agreement	Turns that either express almost the exact thing in one of the preceding four coded student turns OR affirm the previous statement with a short response like "yeah" or "I agree with what she said.". When a turn seems like it should be coded as an Extension but lacks two clear key terms or ideas, it is likely to be coded as an Agreement.

Table 1: Definitions of the collaboration codes.

Code	Definition
Claim	An arguable statement that presents a particular interpretation of a text or topic. DOES: often (but not always) precedes evidence and warrants. States something that can more or less be contested—infers, predicts, hypothesizes, considers possibilities. DOES NOT: simply recount details from text that are accessible to all readers (everyone knows Macbeth became king)
Evidence	Talk used to support, justify, or back a claim. DOES: includes facts, textual references, anecdotes. Often (but not always) follows a claim. Always proximal to a claim (within 1 or 2 turns) . DOES NOT: does not exist without a claim.
Warrant	Move that provides explanation for why evidence supports the claim. DOES: Always proximal to evidence supporting a claim (almost always follows evidence). DOES NOT: It rarely occurs before claim/ evidence that it is explaining.

Table 2: Definitions of the argumentation codes.

Turn	Speaker	Talk	Collaboration	Argumentation
1	St 22	I think it's completely understandable, obviously because of what happened on his father's final day. But I feel like he doesn't deserve necessarily to feel guilty	New Idea	Claim
		because he was put through so much. Whenever you're in that situation, he's been worn down so much and everything has been taken from him. I feel like in that moment, he couldn't really think of anything he could do because he's already done so much, and so many people are telling him like, "There's nothing you can do."		Evidence
		I don't necessarily think he deserves to feel guilty, but I understand why he would.		Warrant
2	St 20	I agree with St 22. He shouldn't feel guilty because it's not his fault. But at the same time, you can't control how you feel. I guess, that's it.	Agreement	Claim
3	Teacher	When he asked himself about, did he pass the test about Rabbi Eliahu, do you guys think that he passed the test or he failed the test, in your opinion?		
4	St 3	I can almost say he passed the test, in a sense.	New Idea	Claim
		But you have to consider that whatever his father thinks [...]. He never wanted to lose his father. He always tried to help his father until the last moment. But then he was in shock. I feel like in general, he passed the test.		Evidence
5	St 6	Yeah	None	
6	St 1	Sorry, go ahead	None	
7	St 6	Okay. I think a big difference between the rabbi and his son, and Elie is that the rabbi's son acted on it and he deliberately did it. But Elie only had a subconscious thought about it and he never really intended on acting on it. He still gave his rations to him. He didn't take him away. He still felt bad. He tried to protect his father as best he could. He never really wanted him to die. It was more something he thought in the moment. Again, the cancer was getting to his head, too.	Challenge	Claim
		I think he passed his test. I don't think it's a big issue if you just thought about it for a second.		Warrant
...				
11	St 1	Speaking on that note, someone mentioned talking about the "Free at last" part. The way I interpret it personally was that I thought that he felt his father was also free at last because he didn't have to deal with his suffering, which also shows that he did pass the test.	New Idea	Claim
12	St 13	Yeah. I also think whenever Elie talks about his father being a burden, it might not be he feels that his father coming around with him, brings him down,	Extension	Claim
		which I think it certainly does when he was thinking about that on the run. But I think that going back to Robbie's point, I think that it also could mean burden of his father's state and how his father is probably going to die is probably a burden on him mentally, as well as how his father is maybe making his chance to death.		Evidence

Table 3: A sample transcript with annotations for students' turns from DT_22 (T1.5.DT_2022.1.Night).

	Annotation	**DT_19**		**DT_22**	
		Count	Percentage	Count	Percentage
Collaboration	New Idea	802	24.59%	1585	20.87%
	Extension	1014	31.09%	2584	34.03%
	Agreement	38	1.17%	177	2.33%
	Challenge	271	8.31%	401	5.28%
	None	1136	34.84%	2847	37.49%
	Total	3261	100.00%	7594	100.00%
Argumentation	Claim	2054	65.31%	4724	62.99%
	Evidence	764	24.29%	1922	25.63%
	Warrant	327	10.40%	854	11.39%
	Total	3145	100.00%	7500	100.00%

Table 4: Descriptive statistics of the two corpora: DT_19 and DT_22.

Approach	**Prompt**

LLM-multi	Below are the definitions of 4 collaboration classes: New Idea, Extension, Challenge, and Agreement.
	# Definition of the 4 collaboration classes
	New Idea: {Definition of New Idea}
	Extension: {Definition of Extension}
	Challenge: {Definition of Challenge}
	Agreement: {Definition of Agreement}
	You are given a 5-turn conversation in a multi-party classroom discussion. Using the provided definition, your task is to classify the last turn into New Idea, Extension, Challenge, Agreement, or None if it does not belong to the four mentioned classes.
	# Example 1 {Example conversation 1}
	Output (New Idea, Extension, Challenge, Agreement, or None): {gold standard answer}
	...
	# Example 10
	{Example conversation 10}
	Output (New Idea, Extension, Challenge, Agreement, or None): {gold standard answer}
	# Your task
	{5-turn conversation}
	Output (New Idea, Extension, Challenge, Agreement, or None):

LLM-binary	Below are the definitions of 4 collaboration classes: New Idea, Extension, Challenge, and Agreement.
	# Definition of the 4 collaboration classes
	New Idea: {Definition of New Idea}
	Extension: {Definition of Extension}
	Challenge: {Definition of Challenge}
	Agreement: {Definition of Agreement}
	You are given a 5-turn conversation in a multi-party classroom discussion. Using the provided definitions, your task is to identify if the last turn is {One targeted class (New Idea, Extension, Challenge, or Agreement)}. Only answer yes or no.
	# Example 1
	{Example conversation 1}
	Output (yes/no): {gold standard answer}
	...
	# Example 10
	{Example conversation 10}
	Output (yes/no): {gold standard answer}
	# Your task
	{5-turn conversation}
	Output (yes/no):

Table 5: Prompts used for collaboration classification. { } is a placeholder. Definitions of collaboration classes are from Table 1.

Approach	Prompt

All	Below are the definitions of 3 argumentation classes: Claim, Evidence, and Warrant.
	# Definition of the 3 argumentation classes
	Claim: {Definition of Claim}
	Evidence: {Definition of Evidence}
	Warrant: {Definition of Warrant}

LLM-auto	You are given a 5-turn conversation in a multi-party classroom discussion. Using the provided definitions, your task is to segment the last turn into one or more of the following argumentation components: Claim, Evidence, and Warrant. The segmentation must include at least one of these components, but it is not required to include all three. Every word in the last turn must belong to one category. Format your output as follows: Output Claim: { } Evidence: { } Warrant: { } # Example 1 (C) {Example conversation 1} Output Claim: {gold standard claim} Evidence: {gold standard evidence} Warrant: {gold standard warrant} ... # Example 10 (E, W, C) {Example conversation 10 with gold standard output} # Your task {5-turn conversation} Output
LLM-gen	You are given a 5-turn conversation in a multi-party classroom discussion. Using the provided definitions, your task is to segment the last turn into {one specific combination of Claim, Evidence, and Warrant}. Every word in the last turn must belong to one category. Format your output as follows: Output (Optional) Claim: { } (Optional) Evidence: { } (Optional) Warrant: { } # Example 1 {one specific combination of Claim, Evidence, and Warrant} {Example conversation 1} Output (Optional) Claim: {gold standard claim} (Optional) Evidence: {gold standard evidence} (Optional) Warrant: {gold standard warrant} ... # Example 10 {Example conversation 10 with gold standard output} # Your task {5-turn conversation} Output
LLM-judge	You are given a 5-turn conversation in a multi-party classroom discussion and different ways to segment the last turn to Claim, Evidence, and Warrant based on the provided definitions. Your task is to pick the most reasonable segmentation. Answer only one number between 1 and 6. Options: 1. {(C) segmentation} 2. {(E) segmentation} ... 6. {(C, E, W) segmentation} The best option is (a number between 1 and 6):
LLM-acc	You are given a 5-turn conversation in a multi-party classroom discussion. Using the provided definition, your task is to classify the last turn into Claim, Evidence, or Warrant. # Example 1 {Example conversation 1} Output (Claim, Evidence, or Warrant): {gold standard answer} ... # Example 10 {Example conversation 10 with gold standard answer} # Your task {5-turn conversation} Output (Claim, Evidence, or Warrant):

Table 6: Prompts used for argument identification. { } is a placeholder. Definitions of argumentation classes are from Table 2. All approaches share the first row to provide the definitions of the classes to the LLM.

Model	DT_19		DT_22	
	Arg vs Non-arg	All 5 labels	Arg vs Non-arg	All 5 labels
BERT	79.6	65.9	79.1	66.8
LLM-multi	80.1	69.1*	80.5	69.9*
LLM-binary	**84.1**	**73.7***	**86.1***	**73.5***

Table 7: Macro (unweighted) F_1 scores of the Collaboration classification task on the two DT corpora. Bold numbers highlight the best results. * means the number is statistically significant compared to its counterpart in the BERT baseline ($p < 0.05$) based on a Wilcoxon signed-rank test.

Model	DT_19	DT_22
BERT	62.3	62.8
LLM-multi	65.5*	68.2*
LLM-binary	**69.8***	**70.2***

Table 8: Cohen's kappa of the Collaboration classification task on the two DT corpora on all 5 labels. Bold numbers highlight the best results. * means the number is statistically significant compared to its counterpart in the BERT baseline ($p < 0.05$) based on a Wilcoxon signed-rank test.

Label	DT_19			DT_22		
	BERT	**LLM-multi**	**LLM-binary**	**BERT**	**LLM-multi**	**LLM-binary**
New Idea	58.2	61.3*	**65.8***	57.1	61.2*	**64.2***
Extension	67.3	74.7*	**79.1***	68.7	74.1*	**79.9***
Challenge	60.5	62.4*	**66.7***	57.3	60.7*	**65.1***
Agreement	70.1	71.5	**79.6***	73.0	72.3	**78.3***
None	73.4	75.6	**77.3***	78.1	**81.3***	80.1
Weighted F_1	67.3	71.3*	**75.1***	69.8	73.7*	**76.3***

Table 9: F_1 score for each collaboration class on DT_19 and DT_22 data. * means the number is statistically significant compared to its counterpart in the BERT model based on a Wilcoxon signed-rank test. Bold numbers highlight the best results for each label per dataset.

Model	B-C	I-C	B-E	I-E	B-W	I-W	Weighted F_1
BERT-BIO	61.5	73.2	68.3	75.7	60.6	69.3	68.6
LLM-auto	66.4	81.2	70.2	81.2	**64.3**	73.4	**71.4**
LLM-refine	**67.3**	**83.1**	**71.9**	**85.4**	62.3	**76.3**	**73.3**

Table 10: Per-label F_1 scores and average weighted F_1 scores of the argument identification task on DT_22. The labels are B/I-Arg, where B/I represents Beginning/Inside and Arg represents one of the three classes: Claim (C), Evidence (E), Warrant (W). Bold numbers show the best results for each label. All numbers are statistically significant compared to their counterparts in the BERT-BIO ($p < 0.05$), as determined by a Wilcoxon signed-rank test.

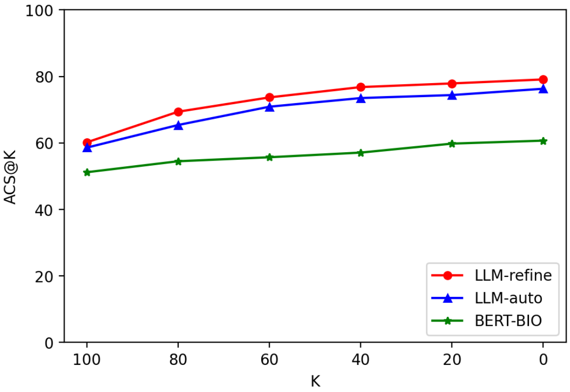

Figure 2: ACS@K with different values of threshold K.

References

Yamen Ajjour, Wei-Fan Chen, Johannes Kiesel, Henning Wachsmuth, and Benno Stein. 2017. Unit segmentation of argumentative texts. In *Proceedings of the 4th Workshop on Argument Mining*, pages 118–128, Copenhagen, Denmark. Association for Computational Linguistics.

Tariq Alhindi and Debanjan Ghosh. 2021. "sharks are not the threat humans are": Argument component segmentation in school student essays. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 210–222, Online. Association for Computational Linguistics.

Edith Bouton and Christa SC Asterhan. 2023. In pursuit of a more unified method to measuring classroom dialogue: The dialogue elements to compound constructs approach. *Learning, Culture and Social Interaction*, 40:100717.

Tuhin Chakrabarty, Christopher Hidey, and Kathy McKeown. 2019. IMHO fine-tuning improves claim detection. 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 558–563, Minneapolis, Minnesota. Association for Computational Linguistics.

Guiming Hardy Chen, Shunian Chen, Ziche Liu, Feng Jiang, and Benyou Wang. 2024a. Humans or LLMs as the judge? a study on judgement bias. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8301–8327, Miami, Florida, USA. Association for Computational Linguistics.

Guizhen Chen, Liying Cheng, Anh Tuan Luu, and Lidong Bing. 2024b. Exploring the potential of large language models in computational argumentation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2309–2330, Bangkok, Thailand. Association for Computational Linguistics.

Michele Luca Contalbo, Francesco Guerra, and Matteo Paganelli. 2024. Argument relation classification through discourse markers and adversarial training. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 18949–18954, Miami, Florida, USA. Association for Computational Linguistics.

Johannes Daxenberger, Steffen Eger, Ivan Habernal, Christian Stab, and Iryna Gurevych. 2017. What is the essence of a claim? cross-domain claim identification. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2055–2066, Copenhagen, Denmark. Association for Computational Linguistics.

Mamoru Deguchi and Kazunori Yamaguchi. 2019. Argument component classification by relation identification by neural network and TextRank. In *Proceedings of the 6th Workshop on Argument Mining*, pages 83–91, Florence, Italy. 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.

Jesús Miguel Garcia-Gorrostieta, Aurelio López-López, David Pinto, Vivek Kumar Singh, Aline Villavicencio, Philipp Mayr-Schlegel, and Efstathios Stamatatos. 2018. Argument component classification in academic writings. *J. Intell. Fuzzy Syst.*, 34(5):3037–3047.

Debela Gemechu, Ramon Ruiz-Dolz, and Chris Reed. 2024. ARIES: A general benchmark for argument relation identification. In *Proceedings of the 11th Workshop on Argument Mining (ArgMining 2024)*, pages 1–14, Bangkok, Thailand. 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, Berlin, Germany. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Hidayaturrahman, Emmanuel Dave, Derwin Suhartono, and Aniati Murni Arymurthy. 2021. Enhancing argumentation component classification using contextual language model. *Journal of Big Data*, 8(1):103.

Heather C Hill, Merrie L Blunk, Charalambos Y Charalambous, Jennifer M Lewis, Geoffrey C Phelps, Laurie Sleep, and Deborah Loewenberg Ball. 2008. Mathematical knowledge for teaching and the mathematical quality of instruction: An exploratory study. *Cognition and instruction*, 26(4):430–511.

Hui Huang, Xingyuan Bu, Hongli Zhou, Yingqi Qu, Jing Liu, Muyun Yang, Bing Xu, and Tiejun Zhao. 2025. An empirical study of llm-as-a-judge for llm evaluation: Fine-tuned judge model is not a general substitute for gpt-4. *Preprint*, arXiv:2403.02839.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Omid Kashefi, Sophia Chan, and Swapna Somasundaran. 2023. Argument detection in student essays under resource constraints. In *Proceedings of the 10th Workshop on Argument Mining*, pages 64–75, Singapore. Association for Computational Linguistics.

Sean Kelly, Andrew M Olney, Patrick Donnelly, Martin Nystrand, and Sidney K D'Mello. 2018. Automatically measuring question authenticity in real-world classrooms. *Educational Researcher*, 47(7):451–464.

Bruce E Larson. 2000. Classroom discussion: a method of instruction and a curriculum outcome. *Teaching and Teacher Education*, 16(5):661–677.

John Lawrence and Chris Reed. 2020. Argument mining: A survey. *Computational Linguistics*, 45(4):765–818.

Jionghao Lin and Kenneth R. Koedinger. 2024. Haror: A system for highlighting and rephrasing open-ended responses. In *Proceedings of the Eleventh ACM Conference on Learning @ Scale*, L@S '24, page 553–555, New York, NY, USA. Association for Computing Machinery.

Luca Lugini and Diane Litman. 2018. Argument component classification for classroom discussions. In *Proceedings of the 5th Workshop on Argument Mining*, pages 57–67, Brussels, Belgium. Association for Computational Linguistics.

Luca Lugini and Diane Litman. 2020. Contextual argument component classification for class discussions. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1475–1480, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Luca Lugini, Christopher Olshefski, Ravneet Singh, Diane Litman, and Amanda Godley. 2020. Discussion tracker: Supporting teacher learning about students' collaborative argumentation in high school classrooms. In *Proceedings of the 28th International Conference on Computational Linguistics: System Demonstrations*, pages 53–58, Barcelona, Spain (Online). International Committee on Computational Linguistics (ICCL).

Lindsay Clare Matsumura, Helen E Garnier, Sharon Cadman Slater, and Melissa D Boston. 2008. Toward measuring instructional interactions "at-scale". *Educational Assessment*, 13(4):267–300.

Jean-Christophe Mensonides, Sébastien Harispe, Jacky Montmain, and Véronique Thireau. 2019. Automatic detection and classification of argument components using multi-task deep neural network. In *Proceedings of the 3rd International Conference on Natural Language and Speech Processing*, pages 25–33, Trento, Italy. Association for Computational Linguistics.

Hiroki Nakayama. 2018. seqeval: A python framework for sequence labeling evaluation. Software available from https://github.com/chakki-works/seqeval.

Vlad Niculae, Joonsuk Park, and Claire Cardie. 2017. Argument mining with structured SVMs and RNNs. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 985–995, Vancouver, Canada. Association for Computational Linguistics.

Christopher Olshefski, Luca Lugini, Ravneet Singh, Diane Litman, and Amanda Godley. 2020. The discussion tracker corpus of collaborative argumentation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1033–1043, Marseille, France. European Language Resources Association.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Axel Pichler, Janis Pagel, and Nils Reiter. 2025. Evaluating LLM-prompting for sequence labeling tasks in computational literary studies. In *Proceedings of the 9th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature (LaTeCH-CLfL 2025)*, pages 32–46, Albuquerque, New Mexico. Association for Computational Linguistics.

Peter Potash, Alexey Romanov, and Anna Rumshisky. 2017. Here's my point: Joint pointer architecture for argument mining. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1364–1373, Copenhagen, Denmark. Association for Computational Linguistics.

Alina Reznitskaya and Maughn Gregory. 2013. Student thought and classroom language: Examining the mechanisms of change in dialogic teaching. *Educational Psychologist*, 48(2):114–133.

Alina Reznitskaya and Ian A.G. Wilkinson. 2021. The argumentation rating tool: Assessing and supporting teacher facilitation and student argumentation during text-based discussions. *Teaching and Teacher Education*, 106:103464.

Claudia Schulz, Steffen Eger, Johannes Daxenberger, Tobias Kahse, and Iryna Gurevych. 2018. Multi-task learning for argumentation mining in low-resource settings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 35–41, New Orleans, Louisiana. Association for Computational Linguistics.

Claudia Schulz, Christian M. Meyer, and Iryna Gurevych. 2019. Challenges in the automatic analysis of students' diagnostic reasoning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6974–6981.

Isabel Segura-Bedmar, Paloma Martínez, and María Herrero-Zazo. 2013. SemEval-2013 task 9 : Extraction of drug-drug interactions from biomedical texts (DDIExtraction 2013). In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 341–350, Atlanta, Georgia, USA. Association for Computational Linguistics.

Tsukasa Shiota and Kazutaka Shimada. 2022. Annotation and multi-modal methods for quality assessment of multi-party discussion. In *Proceedings of the 36th Pacific Asia Conference on Language, Information and Computation*, pages 175–182, Manila, Philippines. Association for Computational Linguistics.

Manfred Stede and Jodi Schneider. 2019. *Argumentation mining*. Springer.

Abhijit Suresh, Jennifer Jacobs, Charis Clevenger, Vivian Lai, Chenhao Tan, James H Martin, and Tamara Sumner. 2021. Using ai to promote equitable classroom discussions: The talkmoves application. In *International Conference on Artificial Intelligence in Education*, pages 344–348.

Nhat Tran and Diane Litman. 2021. Multi-task learning in argument mining for persuasive online discussions. In *Proceedings of the 8th Workshop on Argument Mining*, pages 148–153, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Nhat Tran, Benjamin Pierce, Diane Litman, Richard Correnti, and Lindsay Clare Matsumura. 2024a. Analyzing large language models for classroom discussion assessment. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 500–510, Atlanta, Georgia, USA. International Educational Data Mining Society.

Nhat Tran, Benjamin Pierce, Diane Litman, Richard Correnti, and Lindsay Clare Matsumura. 2024b. Multi-dimensional performance analysis of large language models for classroom discussion assessment. *Journal of Educational Data Mining*, 16(2):304–335.

Henning Wachsmuth, Khalid Al-Khatib, and Benno Stein. 2016. Using argument mining to assess the argumentation quality of essays. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1680–1691, Osaka, Japan. The COLING 2016 Organizing Committee.

Deliang Wang and Gaowei Chen. 2024. On the interpretability of deep learning models for collaborative argumentation analysis in classrooms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 506–516, Bangkok, Thailand. Association for Computational Linguistics.

Liang Wang, Nan Yang, and Furu Wei. 2024. Learning to retrieve in-context examples for large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1752–1767, St. Julian's, Malta. Association for Computational Linguistics.

Rose Wang and Dorottya Demszky. 2023. Is chatgpt a good teacher coach? measuring zero-shot performance for scoring and providing actionable insights on classroom instruction. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pages 626–667.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, Guoyin Wang, and Chen Guo. 2025. GPT-NER: Named entity recognition via large language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4257–4275, Albuquerque, New Mexico. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Paiheng Xu, Jing Liu, Nathan Jones, Julie Cohen, and Wei Ai. 2024. The promises and pitfalls of using language models to measure instruction quality in education. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4375–4389.