Encoders Strike Back: Why and How a Strong Discriminative Baseline Outperforms LLMs in Argument Mining Across Three Heterogeneous Datasets

Fatma-Zohra Rezkellah[1] **Ghassen Karray**[2] **Reto Gubelmann**[1]
[1]University of Zurich (UZH), [2]Independent
{fatma-zohra.rezkellah, ghassen.karray, reto.gubelmann}@uzh.ch

Abstract

We present a comprehensive study of LLMs for argument mining, comparing four learning paradigms against a strong encoder-only baseline across three heterogeneous corpora to assess cross-domain generalisation. We evaluate in-context learning, supervised fine-tuning (SFT), and cold-start reinforcement fine-tuning (RFT) via GRPO. Our results demonstrate that BERT-style encoder-only transformers consistently outperform much larger decoder-only LLMs across all settings. Further mechanistic analysis reveals that the central bottleneck is not knowledge but expression: linear probing of frozen hidden states recovers argument relations at 69–80% accuracy from base models, yet edge generation remains poor across all paradigms. Attention analysis confirms a universal structural blindness across 27 tested conditions.

1 Introduction

When a scientist reviews a clinical trial, a judge weighs competing arguments, or a citizen evaluates a policy debate, they engage in reasoning that is at once structured and contested. The NLP task of Argument Mining (AM) (Lawrence and Reed, 2020) aims to recover this structure automatically: given a text, extract a directed graph whose nodes are argumentative discourse units (ADUs) and whose edges encode support or attack relations, providing a computational foundation for studying how humans reason, disagree, and construct knowledge.

What makes AM a particularly demanding testbed for large language models (LLMs) is its dual challenge. First, it is inherently *perspectival*: depending on one's overall worldview, one's representation of argumentative structure and even judgments about text quality can vary substantially; reduction to one single ground truth is often oversimplifying to the point of distorting (Abercrombie et al., 2025; Teruel et al., 2018), and trained annotators systematically disagree (Davani et al., 2022; Hautli et al., 2022).

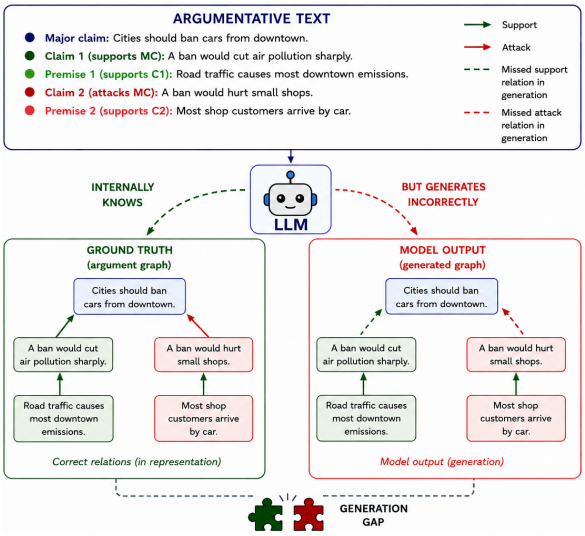


Figure 1: The generation gap in argument graph prediction. Given an argumentative text, the LLM internally encodes the correct support and attack relations (left), yet fails to externalise them coherently during autoregressive graph generation (right), producing an incorrect argument structure.

Model performance is therefore bounded not only by capability but by the complexity of the annotation scheme itself (Castro, 2021; Fleisig et al., 2023). Second, it requires *global graph inference*: a well-formed argument graph over N spans demands coordinated decisions about $O(N^2)$ potential edges, each dependent on all the others, yet autoregressive generation resolves them sequentially and irrevocably, one token at a time.

We frame end-to-end argument graph generation as a unified structured prediction task and compare four paradigms: a discriminative BERT-family pipeline, in-context learning (ICL), supervised fine-tuning (SFT), and cold-start reinforcement fine-tuning (RFT) via GRPO. Beyond benchmarking, we pursue a mechanistic investigation through representation and attention probing to diagnose where

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 50696–50713
October 24-29, 2026 ©2026 Association for Computational Linguistics

models succeed and fail. Our findings reframe the central challenge. It is not a lack of internally represented argument structure, but rather a coordinated framing of what they represent into globally consistent graph output.

Contributions. (i) A sophisticated encoder-only baseline that sets a new state of the art and outperforms all tested LLM configurations; (ii) A broad, multi-methodological study of LLMs' potential in argument mining, including the first application of Reinforcement Learning (RL) to end-to-end argument graph generation, with reward ablations across four formulations; (iii) Representation and attention probing decomposing the performance gap into representational and generative components.

2 Related Work

Argument Mining. AM studies the identification of argumentative components and their relations (Peldszus and Stede, 2015; Lippi and Torroni, 2016). Stab and Gurevych (2017) introduced the persuasive essays benchmark (AAE), while later work extended AM to legal and biomedical domains (Mochales and Moens, 2011; Mayer et al., 2020).

Argument Component Identification. Early ACI systems relied on hand-crafted features (Mochales and Moens, 2011; Stab and Gurevych, 2014; Peldszus and Stede, 2015), later replaced by BiLSTM-CRF and transformer-based BIO tagging (Eger et al., 2017). Recent work reframes ACI as a generative task using instruction-tuned LLMs, though large-scale evaluations reveal persistent lexical shortcutting and weak cross-domain generalisation (Feger et al., 2025).

Argument Relation Classification. Early ARC models jointly predicted relations, and components (Stab and Gurevych, 2017); subsequent work showed many systems rely on positional heuristics rather than semantic understanding (Opitz and Frank, 2019). Neural methods incorporated logical and causal signals (Jo et al., 2021), and recent approaches combine LMs with knowledge graph embeddings (Becker et al., 2020).

LLMs for AM. LLMs have enabled unified prompt-based formulations (Li et al., 2025; Abkenar et al., 2024), joint component–relation prediction (Cabessa et al., 2025; Savigny and Yun, 2026), and end-to-end argument graph generation (Clayton, 2026). Prompt-based methods reformulate ARC as reading comprehension (Bao et al., 2022) or apply chain-of-thought reasoning (Ren et al., 2026). Related unified structure prediction approaches (Morio et al., 2022; Kawarada et al., 2024; Bao et al., 2025) do not investigate RL. Encoder-only models remain competitive in precision-critical settings (Held and Habernal, 2025), and cross-domain generalisation remains unresolved (Feger et al., 2025).

3 Task and Datasets

Task formalization. Given a document d, we define an argument graph $G_d = (V_d, E_d, \tau_d, \lambda_d)$, where V_d are ADU nodes, $\tau_d : V_d \to \mathrm{Substr}(d)$ maps each node to a contiguous text span, $E_d \subseteq V_d \times V_d$ are directed edges between those spans, and $\lambda_d : E_d \to \{-1, +1\}$ assigns support or attack labels. The task $f : d \mapsto G_d$ jointly performs component identification and relation extraction. Since our three corpora use incompatible annotation systems (Toulmin-style (Toulmin, 2003), IAT (Reed and Budzynska, 2011), and PICO-based (Nye et al., 2018)), we normalize all ADU types to a single claim class and all relations to the binary support/attack distinction, to enable a controlled cross-corpus comparison under a unified protocol.

Datasets. Figure 2 illustrates samples from each of the three corpora. **Argument Annotated Essays (AAE)** (Stab and Gurevych, 2017) consists of persuasive essays with premise, claim, and major-claim annotations; its monological structure yields relatively regular argument graphs. **ChangeMyView (CMV)** (Hidey et al., 2017) is sourced from Reddit threads, featuring multi-speaker dialogue, long discourse chains (18+ ADUs per document), and lower inter-annotator agreement, making it the hardest corpus. **AbstRCT** (Mayer et al., 2020) contains RCT abstracts with a highly regular rhetorical structure (background–evidence–conclusion) that makes argument topology more predictable. Table 1 summarizes the three corpora.

Corpus	Docs	IAA
AAE (Stab and Gurevych, 2017)	402	α_U=0.72
CMV (Hidey et al., 2017)	75	α=0.57
AbstRCT (Mayer et al., 2020)	670	κ=0.72

Table 1: Data statistics. IAA = inter-annotator agreement.

All corpora use a fixed 70/30 train/test split, held

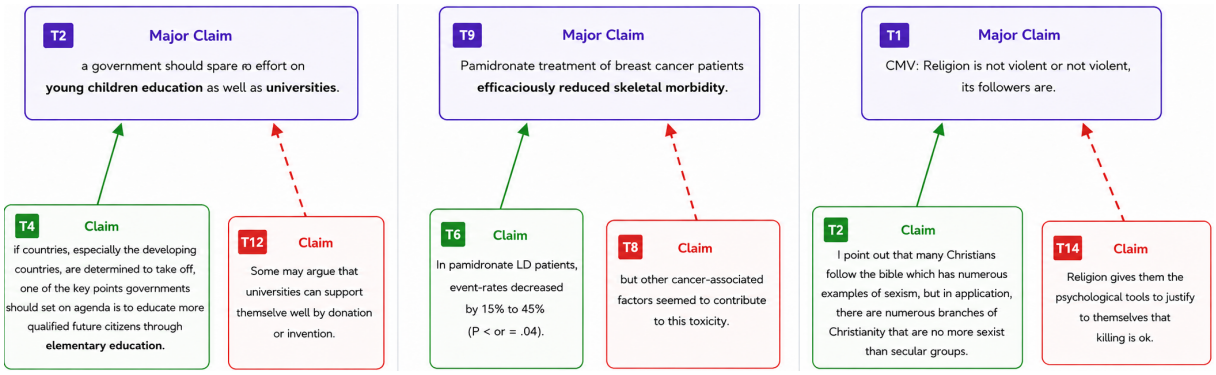


Figure 2: Argument structures in AAE (left), AbstRCT (center), CMV (right). Each panel presents one major claim and two related claims. The green solid arrow denotes support; the red dashed arrow denotes attack.

constant across all four paradigms to ensure direct, unconfounded cross-paradigm comparison. Figure 3 reports the resulting distribution of gold ADU types and relation labels for each corpus. AAE and AbstRCT are heavily support-skewed (94.3% and 87.1% of edges, respectively), consistent with their more monological or formulaic rhetorical structure, while CMV's dialogic, multi-party structure yields a substantially more balanced relation distribution (75.5% support, 24.5% attack).

4 Methods

4.1 Discriminative Encoder (DE)

We implement a two-stage pipeline using BERT-base-uncased (Devlin et al., 2019), RoBERTa-base (Liu et al., 2019), and ModernBERT-base (Warner et al., 2025). **Stage 1 (Component Identification)** finds argumentative spans by assigning each token a BIO label (Beginning, Inside, or Outside a span) via a linear classification head over encoder representations. Since argumentative tokens are rare compared to background text, we use class-weighted cross-entropy to prevent the model from trivially predicting non-argumentative tokens everywhere. **Stage 2 (Relation Classification)** takes every ordered pair of extracted spans (a_i, a_j), encodes them as a single sequence `[CLS]` a_i `[SEP]` a_j `[SEP]`, and classifies the pair as *support*, *attack*, or *no relation* using the `[CLS]` representation. Full equations are in Appendix B.

4.2 In-Context Learning (ICL)

We evaluate Llama-3.3-70B and 3.1-8B (Grattafiori et al., 2024), Mistral-24B and 7B (Mistral AI, 2024), and Qwen-2.5-7B (Team, 2024) under $k \in \{0, 3, 6\}$ few-shot settings, with demonstrations sampled as (document, gold graph) pairs. GPT-5.4 (Singh et al., 2025) serves as a strong closed-weight baseline. The full prompt is in Appendix A.

4.3 Supervised Fine-Tuning (SFT)

We fine-tune the open-weight models above using QLoRA (Dettmers et al., 2023), optimizing autoregressive generation of the complete argument graph conditioned only on gold completion tokens. Full implementation details are in Appendix B.3.

4.4 Cold-Start Reinforcement Fine-Tuning (RFT)

We apply reinforcement learning with verifiable rewards (RLVR) via group relative policy optimization (GRPO) (Shao et al., 2024) directly to instruction-tuned checkpoints. Both the initial π_θ and reference policy π_{ref} are initialized from the same checkpoint, so the model must learn output format and argumentative reasoning purely from reward signals (full objective in Appendix B.4). For each input, the policy samples $N{=}8$ rollouts; each predicted graph $\hat{G}$ is scored against the gold graph G^* by:

$$\begin{aligned} r(\hat{G}, G^*) = \mathbf{1}[\text{valid YAML}] \\ + \mathbf{1}[\text{valid graph}] \\ + 2 \cdot \text{F1F1}(\hat{G}, G^*), \in [0, 4] \end{aligned} \quad (1)$$

The reward has three components. The first binary term $\mathbf{1}[\text{valid YAML}]$ gives credit if the output is syntactically parseable as YAML. The second binary term $\mathbf{1}[\text{valid graph}]$ gives additional credit if the parsed output constitutes a valid argument graph, i.e., all edge endpoints reference declared nodes and relation labels are drawn from $\{\text{support}, \text{attack}\}$. The third term $\text{F1F1}(\hat{G}, G^*) \in [0, 1]$ is the harmonic mean of Node F1 and Edge F1

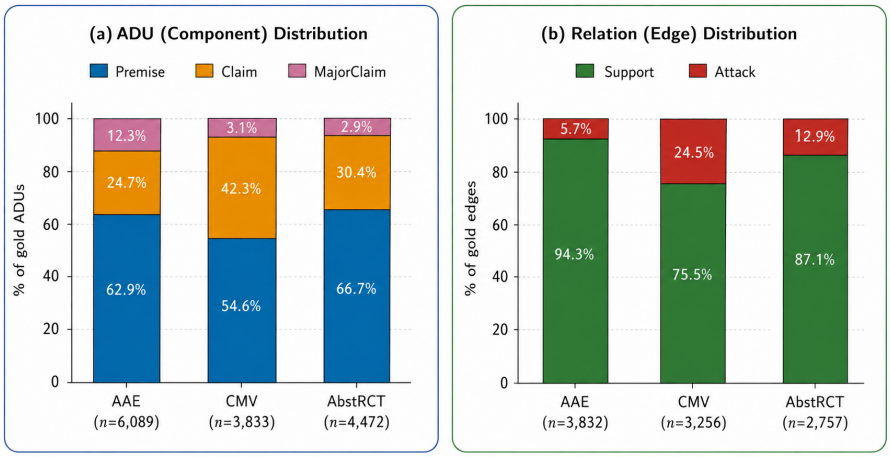


Figure 3: Distribution of gold ADU types and relation labels across the three corpora. Panel (a) reports original annotation types. Panel (b) collapses each corpus's finer-grained relation taxonomy to the binary support/attack distinction used throughout this work.

(Section 5), measuring how closely the predicted graph matches the gold graph at both the component and relational levels; it is weighted by 2 to reflect that semantic quality is the primary training objective. The overall reward is thus bounded in $[0, 4]$, where the binary terms reward structural well-formedness and F1F1 rewards argumentative correctness.

5 Evaluation

We report three metrics. **Node F1** evaluates component identification by matching predicted spans to gold via word-level Jaccard similarity. **Edge F1** evaluates relational structure by matching predicted argument triplets (source, target, relation label) to gold, requiring agreement on both spans and relation type. **F1F1** is the primary metric: the harmonic mean of Node F1 and Edge F1, which penalizes imbalance between component and relation performance and provides a holistic assessment of argument graph quality (Details in Appendix B.5). Results are over four runs with different seeds to account for the non-deterministic nature of LLMs.

6 Results

6.1 Discriminative Models

Table 2 shows in-domain results, which, to the best of our knowledge, exceed previously reported encoder-only performance on these corpora. We note that prior work reports a variety of metrics (macro/micro F1 for components and relations separately), whereas we report F1F1 which is a *stricter* joint measure (Section 5). The comparison is therefore conservative: a system that scores well on F1F1 must perform well on both subtasks simultaneously, whereas a system reporting only component F1 or only relation F1 may be strong on one while weak on the other. With this in mind, prior work reports substantially lower scores on the individual subtasks: on AAE, RoBERTa achieves 0.69 macro F1 for relations (Feger et al., 2025) and 71.35 C-F1 for components (Ruiz-Dolz et al., 2021); on CMV, selectively fine-tuned Longformer reaches 0.59 micro F1 for components and 0.69 F1 for relations (Chakrabarty et al., 2019); and on AbstRCT, SciBERT+GRU+CRF achieves 0.89 micro F1 for components while RoBERTa-based models reach 61.19 macro F1 for relations (Mayer et al., 2020). In our approach, BERT-base achieves F1F1 $= 0.89$ on AAE, 0.92 on AbstRCT, and 0.61 on CMV, exceeding the reported inter-annotator agreements on each corpus. Under domain shift, Edge F1 deteriorates more severely than Node F1 across all models; AbstRCT-trained models generalise worst, reflecting the atypical rhetorical conventions of clinical trial discourse (Table 3, see also Appendix Figure 9).

6.2 In-Context Learning

Node F1 consistently exceeds Edge F1 across all models and shot counts, indicating that relation extraction remains beyond the reach of prompting. GPT-5.4 (3-shot) is the strongest overall model with F1F1 $= 0.61$ on AbstRCT, and 0.46 on AAE. Among open-weight models, Llama-3.3-70B achieves the best result on AbstRCT (F1F1 $=$ 0.531), **but still recovering less than 60% of the**

Model	Corpus	Node F1	Edge F1	F1F1
BERT-base	AAE	0.93	0.86	0.89
	CMV	0.79	0.49	0.61
	AbstRCT	0.93	0.91	**0.92**
RoBERTa	AAE	0.86	0.53	0.66
	CMV	0.75	0.42	0.54
	AbstRCT	0.91	0.85	**0.88**
ModernBERT	AAE	0.9	0.84	0.87
	CMV	0.88	0.43	0.58
	AbstRCT	0.95	0.90	**0.92**

Table 2: Discriminative models: in-domain results. Green = best F1F1 per model.

Model	Train	AAE	CMV	AbstRCT
BERT-base	AAE	**0.89**	0.55	0.51
	CMV	0.46	**0.61**	0.40
	AbstRCT	0.42	0.37	**0.92**
RoBERTa	AAE	**0.66**	0.57	0.52
	CMV	0.46	**0.54**	0.41
	AbstRCT	0.44	0.48	**0.88**
ModernBERT	AAE	**0.87**	0.58	0.47
	CMV	0.47	**0.58**	0.39
	AbstRCT	0.38	0.45	**0.92**

Table 3: Discriminative cross-domain F1F1. Rows = train set, columns = test set. Diagonal entries are in-domain.

gap to BERT-base. We give more details and an overview on these results in the Appendix B.2.

6.3 Supervised Fine-Tuning

SFT substantially outperforms ICL, including the closed-source frontier LLM, and narrows the gap to discriminative models, compare the results shown in Table 4. Llama-3.3-70B achieves F1F1 = 0.74 on AbstRCT and Mistral-24B reaches 0.67. Edge F1 remains the primary bottleneck across all models and corpora. Cross-domain performance mirrors discriminative models: **Edge F1 collapses near zero under every train-test shift, with F1F1 below** 0.25 **in all configurations**; the most severe degradation occurs when training on AbstRCT. Full cross-domain results are in Appendix C.

6.4 Cold-Start RFT

Table 5 shows RFT in-domain results. Due to the high computational cost of GRPO training (see Appendix B), RFT experiments are limited to 7–8B-class models. On AAE and CMV, RFT underperforms SFT for all models. On AbstRCT, Qwen achieves F1F1 = 0.524, outperforming its own SFT result (0.504) without any supervised graph training.

Model	Corpus	Node F1	Edge F1	F1F1
Llama-3.3-70B	AAE	0.93	0.60	0.72
	AbstRCT	0.95	0.61	**0.74**
	CMV	0.88	0.21	0.33
Mistral-24B	AAE	0.88	0.54	0.65
	AbstRCT	0.90	0.55	**0.67**
	CMV	0.75	0.23	0.25
Mistral-7B	AAE	0.82	0.49	0.57
	AbstRCT	0.88	0.54	**0.62**
	CMV	0.70	0.19	0.30
Llama-3.1-8B	AAE	0.93	0.49	0.60
	AbstRCT	0.89	0.58	**0.66**
	CMV	0.69	0.07	0.11
Qwen-2.5-7B	AAE	0.81	0.33	0.42
	AbstRCT	0.86	0.43	**0.50**
	CMV	0.80	0.00	0.00

Table 4: SFT in-domain results. Red = anomalous collapse explained in §B.2.

Model	Corpus	Node F1	Edge F1	F1F1
Mistral-7B	AAE	0.620	0.092	0.150
	AbstRCT	0.610	0.324	**0.345**
	CMV	0.449	0.019	0.032
Llama-3.1-8B	AAE	0.780	0.133	0.212
	AbstRCT	0.631	0.377	**0.428**
	CMV	0.569	0.029	0.052
Qwen-2.5-7B	AAE	0.685	0.163	0.238
	AbstRCT	0.781	0.477	**0.524**
	CMV	0.348	0.022	0.038

Table 5: RFT in-domain results.

Figure 4 reveals qualitatively distinct learning trajectories. Qwen is the only model for which reward and F1F1 move together: validation reward climbs from 2.6 to 3.0 while F1F1 rises from 0.32 to 0.52, with non-trivial F1F1 already at epoch 1. Mistral's reward stabilises at ≈2.5 from epoch 8 but F1F1 plateaus at 0.25–0.27 despite continued reward improvement. Llama exhibits high training instability but trends upward, reaching F1F1 = 0.33 by epoch 15.

Reward ablation. Table 6 compares four reward formulations on Qwen × AbstRCT. **Default** (2·F1F1 plus two binary validity terms) achieves the best F1F1 (0.524), with reward and F1F1 moving together throughout training. **Dense** (Node F1 + Edge F1) converges to 0.46 with the most stable gradient signal. **Gated** (IsValidGraph · F1F1) and **Edge-heavy** (0.5·N + 1.5·E) both plateau at 0.41–0.42; Gated exhibits

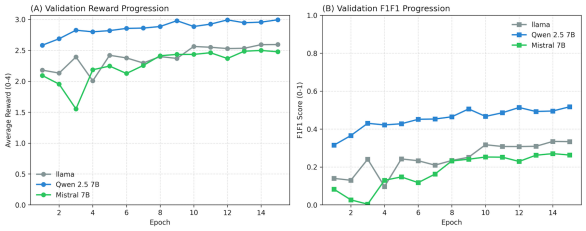


Figure 4: RFT training dynamics on AbstRCT. (A) Validation reward, (B) Validation F1F1. Reward is the Default formulation.

notably slow early learning because its multiplicative structure provides zero gradient when structural validity fails.

Reward	Node F1	Edge F1	F1F1
Default	0.781	0.477	**0.524**
Dense	**0.793**	0.391	0.465
Edge-heavy	0.714	**0.417**	0.420
Gated	0.698	0.413	0.413

Table 6: RFT reward ablation scores

7 Analysis

We conduct a layered analysis moving from surface to mechanism: first, we assess where in the error taxonomy models fail; second, whether the failure is representational or generative. Since Edge F1 is the universal bottleneck across all paradigms and corpora, the analysis focuses primarily on edge prediction errors and is conducted within domain on the 7–8B-class models shared with RFT for direct cross-paradigm comparison.

A natural concern is that models rely on surface discourse markers (DMs) such as *because*, *however*, or *therefore* rather than genuine structural reasoning. However, across all three corpora, 84–85% of gold edges contain no discourse marker (AAE 85.4%, AbstRCT 84.1%, CMV 84.7%), and only 4.0% of attack edges in AAE carry an explicit attack-signalling DM. Furthermore, our analysis reveals that LLMs generally predict edges equally well whether or not a discourse marker is present, with near-zero accuracy differences across all corpora (AAE: +0.016, AbstRCT: +0.008), confirming no linguistic shortcutting. The one exception, SFT on CMV, shows a +0.299 gap.

7.1 Structural Level: Edge Error Taxonomy

We annotate all edge prediction errors with a five-type taxonomy. **Type I** (wrong direction): predicted $A \rightarrow B$, gold has $B \rightarrow A$. **Type II** (wrong label): direction correct, support/attack flipped. **Type III** (spurious edge): both endpoints match gold nodes but no gold edge exists. **Type IV** (missing edge): a gold edge was never predicted. **Type V** (cascaded node error): at least one endpoint cannot be matched to any gold node.

Table 7 reports the full distributions. Four patterns stand out. Type IV dominates universally, accounting for 34–58% of all errors across every paradigm and corpus. Models identify nodes but fail to connect them, a signature of sequential generation without global graph planning. Type III decreases from SFT to RFT (29–39% to 18–35%). The GRPO reward penalizes false-positive edges, driving models toward a conservative strategy: when uncertain, omit rather than hallucinate. Type V rises sharply in RFT: Llama-RFT on AbstRCT shows 43.5% cascaded errors versus 23.4% for Llama-SFT. The reward incentivizes high node recall, but extra spans are noisy and propagate errors to every incident edge. RFT implicitly trades Type III for Type V. Types I and II are consistently below 10%. When models hypothesize a relation, they typically get direction and label right. The hard problem is not classifying a relation once identified; it is deciding whether one should exist at all.

7.2 Internal-Level Analysis: Probing Representations and Attention

The error taxonomy reveals *how* models fail at argument graph generation; we now investigate *why*. Two competing hypotheses can explain poor task performance:

1. **Representation failure**: the model does not internally encode argumentative relations, so no decoder, however well-trained, could produce the correct graph.

2. **Generation failure**: the model encodes the relevant relational information but cannot externalise it as a well-formed structured output.

Distinguishing these two failure modes is critical for directing future work: hypothesis (1) calls for better pretraining data or architectural changes, while hypothesis (2) calls for better decoding strategies or output-format training. We test each hypothesis with a dedicated probing experiment, applied to base, SFT, and RFT checkpoints to track how each training paradigm affects internal representations. We give an overview on our approach in Figure 5.

Model	Corpus	SFT						RFT					
		F1F1	I%	II%	III%	IV%	V%	F1F1	I%	II%	III%	IV%	V%
Mistral	AAE	0.571	5.2	1.7	31.3	48.5	13.2	0.150	5.5	0.9	18.3	47.7	27.5
	AbstRCT	0.623	1.0	3.4	37.3	37.7	20.5	0.345	1.0	3.3	27.1	34.3	34.3
	CMV	0.246	1.4	0.0	20.3	44.1	34.3	0.032	2.9	0.0	11.8	58.0	27.3
Llama	AAE	0.596	8.2	0.7	31.7	49.4	10.0	0.212	5.7	0.5	29.3	36.8	27.7
	AbstRCT	0.659	0.4	3.1	38.7	34.4	23.4	0.428	1.1	1.5	31.7	22.2	**43.5**
	CMV	0.107	0.0	1.8	21.3	39.6	37.3	0.052	3.0	0.0	17.1	50.3	29.5
Qwen	AAE	0.417	4.8	0.4	29.5	57.5	7.7	0.238	5.9	1.1	25.3	44.7	23.0
	AbstRCT	0.504	2.1	1.2	30.9	43.4	22.3	0.524	1.3	1.0	34.9	40.9	21.9
	CMV	0.000	5.7	0.0	34.3	31.4	28.6	0.038	1.6	0.0	5.4	**76.8**	16.2

Table 7: Edge error taxonomy. I: wrong direction, II: wrong label, III: spurious edge, IV: missing edge, V: cascaded node error.

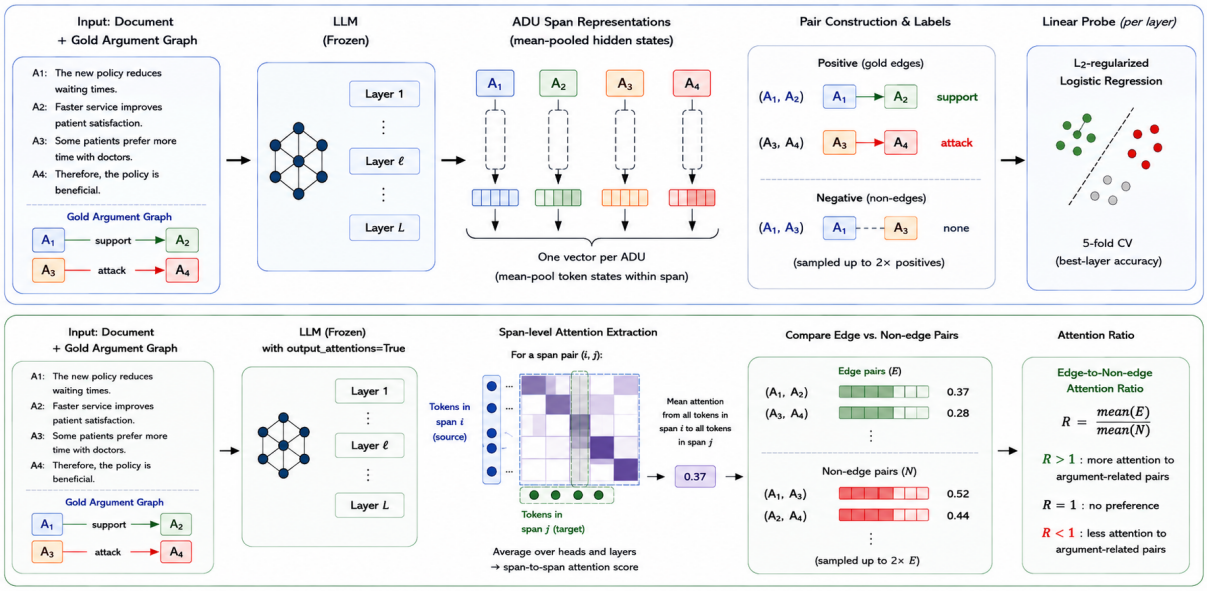


Figure 5: Overview of the probing methodology. Top: Representation probing tests if argumentative relations are encoded in hidden representations. Bottom: Attention probing analyzes whether attention mechanisms preferentially focus on argument-related spans by comparing attention assigned to edge and non-edge ADU pairs.

Representation probing. We adopt the probing methodology of (Belinkov, 2022; Conneau et al., 2018), which trains a lightweight classifier on representations to test whether a target linguistic property is linearly encoded. If a simple probe can recover argument relations from hidden states, the information is *present* regardless of whether the model can generate the correct output.

For each (model, corpus, paradigm) triple we process every validation document through the model and extract hidden states at every fourth transformer layer. Each gold ADU span is aligned to its token positions via Jaccard word-overlap matching and mean-pooled into a single vector. For every gold edge we concatenate the source and target ADU vectors into a *pair representation*; we additionally sample negative pairs: node pairs with no gold edge. Each pair is labelled SUPPORT, ATTACK, or NONE (three-way classification). Negative (NONE) pairs are sampled up to $2\times$ the number of positive pairs per document, both for computational tractability and to avoid a degenerate class distribution in which NONE pairs, which vastly outnumber true relations, dominate the probe's training signal.

At every probed layer we train an ℓ_2-regularised logistic regression probe 5 and report best-layer accuracy in Table 8. Crucially, the probe is linear and trained on frozen representations: high accuracy therefore reflects information already present in the model, not information injected by the probe itself (Hewitt and Liang, 2019). Table 8 reports best-layer probe accuracy. Four observations stand out. *(i) Base models already encode argument relations*

with an accuracy of (69–80 %), confirming that the relational signal predates any task-specific training and is acquired during pretraining. *(ii) SFT changes probe accuracy by at most 0.7 pp*, despite increasing generation-level edge F1 by 20–62 pp. This strongly supports hypothesis (2): SFT teaches the model to *externalise* argument structure it already represents, rather than building completely new representations. RFT checkpoints show the same stability (max change 0.8 pp relative to base), confirming that neither supervised nor reinforcement fine-tuning alters the underlying relational encoding.*(iii) AbstRCT* yields the highest probe accuracy (78–80 %), consistent with the formulaic discourse structure of RCT abstracts. *(iv) Qwen's probe accuracy is within 2–4 pp of Llama and Mistral*, despite substantially lower SFT edge F1, confirming that Qwen's under-performance is a decoder-side effect rather than a knowledge deficit.

Attention probing. Representation probing shows that argument relations are encoded *somewhere* in the hidden states; we now ask whether self-attention heads explicitly *track* argumentative structure by attending preferentially to related ADU spans. Prior work has shown that individual attention heads can specialize in specific linguistic relations such as coreference or syntactic dependency (Clark et al., 2019; Voita et al., 2019); we test whether an analogous specialization emerges for argumentative relations.

For each validation document we run a forward pass with full attention output and extract the attention weight matrices across all layers and heads. For every gold edge and a matched set of non-edge pairs, we compute the mean attention weight flowing between the source and target ADU token spans, averaged across all heads and layers. We summarize the result as an *edge-vs-non-edge attention ratio*: values above 1.0 would indicate that the model attends preferentially to argumentatively related spans.

The ratio ranges from 0.634 to 0.832 across all 27 tested conditions (3 models $\times$ 3 corpora $\times$ 3 paradigms), *universally* below 1.0. Models attend *less* to argument-related span pairs than to unrelated ones, in every configuration. This indicates that argumentative relations are not surfaced through focused span-to-span attention; instead, the relational signal revealed by the representation probe is distributed across the residual stream and accumulated through many individually diffuse attention interactions. Combined with the representation-probing results, this paints a clear picture: argument-relational knowledge is encoded in the hidden states but is *not* mediated by interpretable attention patterns, making it a latent, distributed representation that fine-tuning must learn to decode into explicit output structure.

Model	Paradigm	AAE	AbstRCT	CMV
Qwen	Base	0.692	0.780	0.733
	SFT	0.697	0.770	0.733
	RFT	0.692	0.778	0.714
Llama	Base	0.737	0.791	0.692
	SFT	0.739	0.790	0.697
	RFT	0.737	0.792	0.740
Mistral	Base	0.719	0.801	0.728
	SFT	0.720	0.801	0.735
	RFT	0.723	0.730	0.730

Table 8: Best-layer linear probe accuracy. Stable across paradigms despite large generation differences.

8 Discussion

Our results converge on a diagnosis that cuts across all four paradigms: the central difficulty of end-to-end argument mining is not that models lack knowledge of argument relations, but that the autoregressive generation paradigm cannot coordinate what they know into a globally consistent structured output. It is important to be precise about what we mean by this. Our probing results show that base models already encode *relational* information, specifically, whether two spans stand in a support or attack relation, at 69–80% accuracy from a linear classifier on frozen hidden states. This encoding changes by less than 0.8pp under SFT or RFT, even as generation-level edge F1 improves by 20–62pp. What fine-tuning changes is not the model's internal relational knowledge but its ability to externalise that knowledge into structured output. In other words, fine-tuning instills the correct output format; the relational signal was already there. Generation performance alone may therefore underestimate what the model has encoded: a relation can be recoverable from the model's representations while still failing to appear in the generated graph.

Discriminative encoders don't face this same coordination problem. Their two-stage pipeline separates span detection from relation classification, allowing each decision to condition on the full input without sequential commitment. When a

relation classifier sees the entire document before deciding whether two spans are connected, it can exploit global context freely. An autoregressive decoder making the same decision as one of hundreds of sequential token predictions, unable to revise earlier commitments, is solving a structurally harder problem, not because it knows less, but because the one-pass generation process imposes constraints that the pipeline architecture does not. This is particularly consequential for argument graphs, where multiple edge decisions must ultimately form one globally coherent structure.

Cross-domain generalisation sharpens this picture further. Under every train-test domain shift, edge F1 collapses near zero for all generative paradigms, while discriminative models retain a meaningful fraction of in-domain performance. The reason is that encoder models learn to classify relations between span pairs, a function that depends primarily on the semantic content of the spans themselves and transfers reasonably across domains. Generative models, by contrast, learn a mapping from document to output sequence that is deeply entangled with the surface conventions of the training corpus: specific graph topologies, node orderings, and formatting patterns that do not transfer. RFT degrades less than SFT under domain shift (Llama AbstRCT→AAE: RFT F1F1=0.176 vs. SFT F1F1=0.101), which is consistent with reward-shaped policies being less sensitive to surface format than supervised imitation: the reward signal cares more about correctness, not about reproducing the exact conventions of the training domain. But this is a modest advantage; cross-domain argument graph generation remains largely unsolved across all paradigms.

The behavior of RFT under cold-start conditions illuminates when reward-based optimization can help with this coordination problem and when it cannot. Qwen is the only model for which cold-start RLVR produces notable improvement. The distinguishing factor is that it was itself trained with multistage RL, and an SFT mixture explicitly containing chain-of-thought and structured output data (Team, 2024), it arrives at RL training with a pre-existing reasoning scaffold making cold GRPO a continuation of its existing training regime, and giving it something coherent to refine rather than constructing from scratch. For Llama and Mistral, which are instruction-tuned lacking RL post-training, GRPO must simultaneously teach format, reasoning, and argumentative structure: the signal is too sparse to make progress on all three at once. The path of least resistance is satisfying the binary structural validity terms while leaving F1F1 flat (higher chance of reward hacking). This suggests that reward-based optimization can refine an existing structured-generation capability, but may struggle to construct, from scratch, one from sparse reward alone.

A final dimension that our results expose concerns the nature of evaluation in tasks where the target construct is inherently perspectival. Argument mining asks models to recover structure that trained human annotators themselves disagree on: inter-annotator agreement scores set a ceiling that is independent of model capability. Beyond this ceiling, there is a subtler problem: models may produce outputs that are argumentation-theoretically defensible yet misaligned with the specific choices of a particular annotator. More broadly, recent work has shown that LLM outputs on AM tasks can be logically valid but annotation-scheme misaligned (Elguendouze et al., 2026; Pietroń et al., 2026), treated as errors under evaluation protocols despite reflecting defensible interpretations of discourse structure. As models become more capable, this conflation between modeling error and annotator-framework misalignment will become increasingly difficult to ignore, and the field will need evaluation frameworks that can distinguish between the two.

9 Conclusion

Relation prediction has consistently been the harder challenge in AM, and our results show why: it requires global reasoning over the entire argument graph, a problem autoregressive generation is structurally ill-suited for. The bottleneck is not relational knowledge but the inability to coordinate it under sequential decoding. Pipeline encoder-based models, which sidestep this constraint, remain unmatched by every paradigm/model we test. RLVR seems promising when initialized from models with pre-existing reasoning scaffolds, pointing to reasoning-capable models as the right foundation for future RL-based AM. We close with a question this work cannot fully answer: are remaining errors a limit of current models, or a reflection of the fact that argumentative structure is inherently perspectival and may not permit a single correct answer?

Limitations

This study evaluates RFT only on 7–8B parameter models due to computational constraints; consequently, the scaling behavior of RL tuning remains an open question. Given the improvements typically observed when moving to larger supervised fine-tuned models, it is reasonable to expect that larger post-trained reasoning models could further improve performance under RFT.

To ensure a fair comparison across learning paradigms, all methods were evaluated under the same practical setup, computational budget, and initialization protocol. In particular, RFT experiments were conducted exclusively from base model checkpoints rather than warm-starting from the best supervised fine-tuned models. While this design choice isolates the effect of each training paradigm, warm-start strategies remain a promising direction and could potentially provide stronger results for RFT. Additionally, RL methods are known to exhibit substantially higher sensitivity to hyperparameters and optimization choices than supervised approaches. As a result, the results reported here should be interpreted as corresponding to a single operating point within a considerably larger optimization space. Additional exploration of RFT-specific configurations, including KL regularization tuning, learning schedules, and convergence studies (e.g., required training duration and number of epochs) may further shift the observed performance landscape.

A natural extension is to test whether chain-of-thought prompting narrows the node–edge gap by giving the model room to deliberate before committing to a graph. This direction is conceptually double-edged: If CoT narrows the gap, it would support a generation-failure account in which structured deliberation helps models externalize relations they already encode. But a growing body of work shows that CoT explanations are frequently unfaithful: models can produce fluent, coherent-looking reasoning that does not causally determine the final answer, but is instead constructed to justify a conclusion the model would have reached anyway (Turpin et al., 2023; Lanham et al., 2023). Given that our probing results already show argument relations are linearly decodable from frozen hidden states prior to any reasoning step, it is a live possibility that CoT traces in our setting would be post-hoc rationalizations of relations the model already "decided" in which case CoT would improve surface coherence or output formatting without addressing the underlying coordination failure. Beyond these constraints, we see a genuine opportunity to extend the evaluation methodology itself. A topological evaluation: graph edit distance between predicted and gold graphs, divergence in node in/out-degree distributions, the number of weakly connected components, or counts of cycles in what should be an acyclic structure would characterize *how* predicted graphs deviate from gold beyond triplet-level accuracy. We view this as a promising complement to F1F1, and leave a full topological analysis to future work.

Acknowledgments

We gratefully acknowledge support from project AI-R and the Digital Society Initiative (DSI) at the University of Zurich.

References

Gavin Abercrombie, Valerio Basile, Simona Frenda, Sara Tonelli, and Shiran Dudy, editors. 2025. *Proceedings of the The 4th Workshop on Perspectivist Approaches to NLP*. Association for Computational Linguistics, Suzhou, China.

Mohammad Yeghaneh Abkenar, Weixing Wang, Hendrik Graupner, and Manfred Stede. 2024. Assessing open-source large language models on argumentation mining subtasks. *arXiv preprint arXiv:2411.05639*.

Jianzhu Bao, Mohan Jing, Kuicai Dong, Aixin Sun, Yang Sun, and Ruifeng Xu. 2025. Uniasa: A unified generative framework for argument structure analysis. *Computational Linguistics*, 51(3):739–784.

Jianzhu Bao, Jingyi Sun, Qinglin Zhu, and Ruifeng Xu. 2022. Have my arguments been replied to? argument pair extraction as machine reading comprehension. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 29–35.

Maria Becker, Ioana Hulpuş, Juri Opitz, Debjit Paul, Jonathan Kobbe, Heiner Stuckenschmidt, and Anette Frank. 2020. Explaining arguments with background knowledge: Towards knowledge-based argumentation analysis. *Datenbank-Spektrum*, 20(2):131–141.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Jérémie Cabessa, Hugo Hernault, and Umer Mushtaq. 2025. Argument mining with fine-tuned large language models. In *Proceedings of the 31st international conference on computational linguistics*, pages 6624–6635.

Diego Castro. 2021. Argumentation and persistent disagreement. *Informal Logic*, 41(2):245–280.

Tuhin Chakrabarty, Christopher Hidey, Smaranda Muresan, Kathy McKeown, and Alyssa Hwang. 2019. Ampersand: Argument mining for persuasive online discussions. In *Proceedings of EMNLP*, pages 2933–2943.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. 2019. What does bert look at? an analysis of bert's attention. In *Proceedings of the 2019 ACL workshop BlackboxNLP: analyzing and interpreting neural networks for NLP*, pages 276–286.

Jonathan Clayton. 2026. *Graphical Summarisation of Argumentative Text*. Ph.D. thesis, University of Sheffield.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136.

Aida Mostafazadeh Davani, Mark Díaz, and Vinodkumar Prabhakaran. 2022. Dealing with disagreements: Looking beyond the majority vote in subjective annotations. *Transactions of the Association for Computational Linguistics*, 10:92–110.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *Advances in neural information processing systems*, 36:10088–10115.

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.

Steffen Eger, Johannes Daxenberger, and Iryna Gurevych. 2017. Neural end-to-end learning for computational argumentation mining. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11–22.

Sofiane Elguendouze, Erwan Hain, Elena Cabrio, and Serena Villata. 2026. Compact prompting in instruction-tuned llms for joint argumentative component detection. *arXiv preprint arXiv:2603.03095*.

Marc Feger, Katarina Boland, and Stefan Dietze. 2025. Limited generalizability in argument mining: State-of-the-art models learn datasets, not arguments. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 23900–23915.

Eve Fleisig, Rediet Abebe, and Dan Klein. 2023. When the majority is wrong: Modeling annotator disagreement for subjective tasks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6715–6726.

Aaron Grattafiori and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Annette Hautli, Ella Schad, and Chris Reed. 2022. Disagreement space in argument analysis. In *Proceedings of the 1st Workshop on Perspectivist Approaches to NLP@ LREC2022*, pages 1–9.

Lena Held and Ivan Habernal. 2025. Contemporary llms struggle with extracting formal legal arguments. In *Proceedings of the Natural Legal Language Processing Workshop 2025*, pages 292–303.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. 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 2733–2743.

Christopher Hidey, Elena Musi, Alyssa Hwang, Smaranda Muresan, and Kathleen McKeown. 2017. Analyzing the semantic types of claims and premises in an online persuasive forum. In *Proceedings of the 4th Workshop on Argument Mining*, pages 11–21.

Yohan Jo, Seojin Bang, Chris Reed, and Eduard Hovy. 2021. Classifying argumentative relations using logical mechanisms and argumentation schemes. *Transactions of the Association for Computational Linguistics*, 9:721–739.

Masayuki Kawarada, Tsutomu Hirao, Wataru Uchida, and Masaaki Nagata. 2024. Argument mining as a text-to-text generation task. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2002–2014.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, and 1 others. 2023. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*.

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

Hao Li, Viktor Schlegel, Yizheng Sun, Riza Batista-Navarro, and Goran Nenadic. 2025. Large language models in argument mining: A survey. *arXiv preprint arXiv:2506.16383*.

Marco Lippi and Paolo Torroni. 2016. Argumentation mining: State of the art and emerging trends. *ACM Transactions on Internet Technology (TOIT)*, 16(2):1–25.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Tobias Mayer, Elena Cabrio, and Serena Villata. 2020. Transformer-based argument mining for healthcare applications. In *ECAI 2020: 24th European Conference on Artificial Intelligence, 29 August–8 September 2020, Santiago de Compostela, Spain–Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020)*, pages 2108–2115. SAGE Publications 1 Oliver's Yard, 55 City Road, London, EC1Y 1SP.

Mistral AI. 2024. Mistral models. https://mistral.ai.

Raquel Mochales and Marie-Francine Moens. 2011. Argumentation mining. *Artificial intelligence and law*, 19(1):1–22.

Gaku Morio, Hiroaki Ozaki, Terufumi Morishita, and Kohsuke Yanai. 2022. End-to-end argument mining with cross-corpora multi-task learning. *Transactions of the Association for Computational Linguistics*, 10:639–658.

Benjamin Nye, Junyi Jessy Li, Roma Patel, Yinfei Yang, Iain Marshall, Ani Nenkova, and Byron C Wallace. 2018. A corpus with multi-level annotations of patients, interventions and outcomes to support language processing for medical literature. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 197–207.

Juri Opitz and Anette Frank. 2019. Dissecting content and context in argumentative relation analysis. In *Proceedings of the 6th Workshop on Argument Mining*, pages 25–34.

Andreas Peldszus and Manfred Stede. 2015. Joint prediction in mst-style discourse parsing for argumentation mining. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 938–948.

Marcin Pietroń, Filip Gampel, Jakub Gomułka, Andrzej Tomski, and Rafał Olszowski. 2026. A comprehensive study of llm-based argument classification: from llama through deepseek to gpt-5.2. *Neural Computing and Applications*, 38(11):447.

Chris Reed and Katarzyna Budzynska. 2011. How dialogues create arguments. In *Proceedings of the 7th conference on argumentation of the International Society for the Study of Argumentation*, pages 1633–1645.

Yupei Ren, Ning Zhang, Xiaoyu Li, Yadong Zhang, Yuqing Chen, and Man Lan. 2026. Toward sustainable education: Generative ai-powered argument mining in student writing. *Sustainability*, 18(7):3338.

Ramon Ruiz-Dolz, Jose Alemany, Stella Heras, and Ana Garcia-Fornes. 2021. Transformer-based models for automatic identification of argument relations: A cross-domain evaluation. *IEEE Intelligent Systems*, 36(6):62–70.

Henri Savigny and Bruno Yun. 2026. Amelia: A family of multi-task end-to-end language models for argumentation. *Argument & Computation*, 17(1):79–103.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, and 1 others. 2025. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*.

Christian Stab and Iryna Gurevych. 2014. 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. Parsing argumentation structures in persuasive essays. *Computational linguistics*, 43(3):619–659.

Qwen Team. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Milagro Teruel, Cristian Cardellino, Fernando Cardellino, Laura Alonso Alemany, and Serena Villata. 2018. Increasing argument annotation reproducibility by using inter-annotator agreement to improve guidelines. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

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

Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. 2023. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36:74952–74965.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 5797–5808.

Leandro von Werra and 1 others. 2022. TRL: Transformer reinforcement learning. *GitHub repository*.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom

Aarsen, and 1 others. 2025. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. In *Proceedings of the 63rd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pages 2526–2547.

A Prompt & Output Examples

Figure 6 presents the prompt used across all methods and models, while Figure 7 illustrates the observed collapse of the Qwen model on the CMV dataset.

B Methods

B.1 Discriminative Encoder Training

Task 1: Component Identification. Each token x_t in the input sequence receives a BIO label via a linear head over the encoder output $\mathbf{h}_t$:

$$p(y_t \mid \mathbf{x}) = \text{softmax}(\mathbf{W}\mathbf{h}_t), \quad y_t \in \{\texttt{O}, \texttt{B}, \texttt{I}\} \tag{2}$$

Contiguous B-claim/I-claim spans are then merged into ADUs. To address the potential imbalance toward O tokens, we use a weighted cross-entropy loss:

$$\mathcal{L}_1 = -\frac{1}{T}\sum_{t=1}^{T} w_{y_t} \log p(y_t \mid \mathbf{x}), \quad w_{y_t} = \frac{N}{K \cdot N_{y_t}} \tag{3}$$

where T is the sequence length, $y_t \in \{\texttt{B}, \texttt{I}, \texttt{O}\}$ is the gold label at position t, N is the total number of training tokens, $K = 3$ is the number of classes, and N_{y_t} is the count of tokens belonging to class y_t. The intuition is straightforward: since O tokens vastly outnumber B/I tokens, a uniform loss would cause the model to trivially predict O everywhere. The inverse-frequency weight w_{y_t} corrects for this by assigning higher penalty to mistakes on rare classes: the fewer tokens a class has, the larger its weight, forcing the model to treat argumentative span boundaries as equally important as background tokens despite their scarcity.

Task 2: Relation Classification. Each ordered pair of ADUs (a_i, a_j) extracted by Task 1 is encoded as a single sequence $[\texttt{CLS}]\, a_i\, [\texttt{SEP}]\, a_j\, [\texttt{SEP}]$, and the [CLS] vector $\mathbf{h}_{[\texttt{CLS}]}$ is classified by:

$$p(r \mid a_i, a_j) = \text{softmax}(\mathbf{W}\mathbf{h}_{[\texttt{CLS}]}), \quad r \in \{\texttt{no_rel}, \texttt{support}, \texttt{attack}\} \tag{4}$$

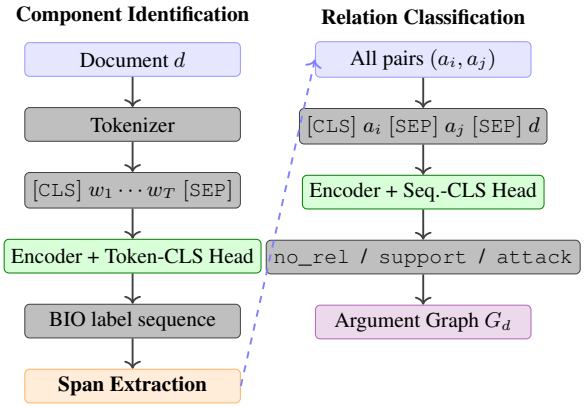


Figure 8: Discriminative pipeline: Beginning-Inside-Outside (BIO) tagging extracts $ADUs$, which feed pairwise relation classification to produce G_d.

Figure 9 gives the difference between node and edge F1 for BERT-base, trained on AAE, tested out-of-domain.

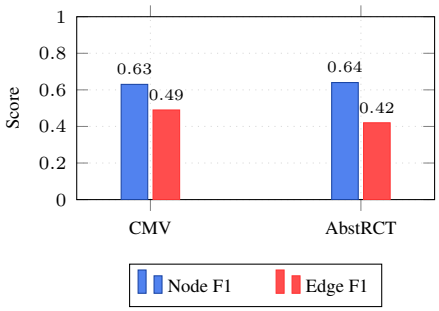


Figure 9: BERT-base trained on AAE, tested out-of-domain: Node F1 vs. Edge F1.

B.2 In-Context Learning (ICL)

Figure 10 gives the results per LLM in their respective best fs-configuration. Shot count does not exhibit a monotonic effect: some configurations benefit from additional demonstrations while others peak at intermediate k and degrade (e.g., Llama-3.3-70B on CMV peaks at k=3 then drops at k=6). Qwen collapses on CMV at all shot settings, a failure that persists identically under SFT and RFT. Inspection of outputs indicates a primarily generative failure mode, characterized by invalid or non-standard node identifiers, repetition loops in node generation, and edge references that fail to correspond to valid nodes (see Fig. 7)

You are an expert Argument Mining system. Your task is to extract the full argumentative structure of a given text as a directed graph. The output must represent: - Argumentative Discourse Units (ADUs) as nodes - Argumentative relations as directed edges

DEFINITIONS

1. Argumentative Discourse Unit (ADU): An ADU is a minimal, self-contained span of text that expresses a single claim or statement that can participate in reasoning.
- Extract ADUs as exact contiguous substrings from the input text - Prefer minimal spans that preserve meaning - Do not merge multiple independent claims into one ADU - Do not split a single coherent claim unnecessarily
2. Relations: Relations are directed and defined between pairs of ADUs.
- Support (relation = 1): ADU A supports ADU B if A provides evidence, justification, or a reason to believe B.
- Attack (relation = -1): ADU A attacks ADU B if A contradicts, refutes, or weakens B.

REASONING GUIDELINES

Before producing the final output, you must internally perform the following steps:
1. Identify all ADUs in the text 2. For each pair of ADUs (A, B), determine whether: - A supports B - A attacks B - or no relation exists 3. Ensure that relations reflect the logical structure of the argument, not just surface proximity Important: - Relations must reflect reasoning, not topic similarity - Avoid adding edges unless a clear argumentative relation exists

GRAPH CONSTRAINTS

The final graph must satisfy:
- Each node corresponds to exactly one unique ADU - No duplicate nodes - No duplicate edges - No self-loops (a node cannot relate to itself) - All edges must connect valid node IDs - The graph must be well-formed and consistent

ALIGNMENT WITH TEXT

- Node texts MUST match exact substrings from the input text - Do NOT paraphrase, summarize, or modify text - Do NOT hallucinate content not present in the input

OUTPUT FORMAT

Your response MUST consist ONLY of a valid YAML object enclosed in triple backticks (Do not include any explanation or text outside the YAML block).The structure is:
“‘yaml
nodes:
- id: <unique_integer>
text: "<exact substring from input>"

edges:
- source: <node_id>
target: <node_id>
relation: <1 for support or -1 for attack> “‘

Figure 6: System prompt.

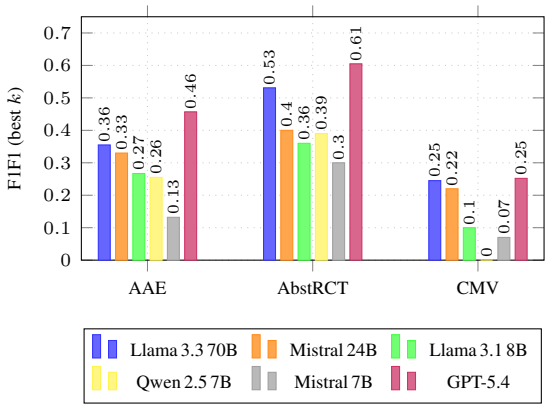


Figure 10: ICL: best F1F1 per model across $k \in \{0, 3, 6\}$ shots.

B.3 Supervised Fine-Tuning (SFT)

We fine-tune LLMs with QLoRA (Dettmers et al., 2023) using the TRL `SFTTrainer` (von Werra et al., 2022):

QLoRA freezes base weights and trains only low-rank adapter matrices, making fine-tuning of 7–70B models feasible on a single GPU. The SFT objective is causal next token prediction loss over gold completions:

$$\mathcal{L}_{\text{SFT}} = -\frac{1}{T}\sum_{t=1}^{T} \log p_\theta(y_t \mid y_{<t},\, d,\, \mathcal{P}_{\text{sys}})$$

where y_t is the t-th token of the ground truth, $y_{<t}$ is the preceding output tokens, d is the input document, and T is the total number of output tokens.

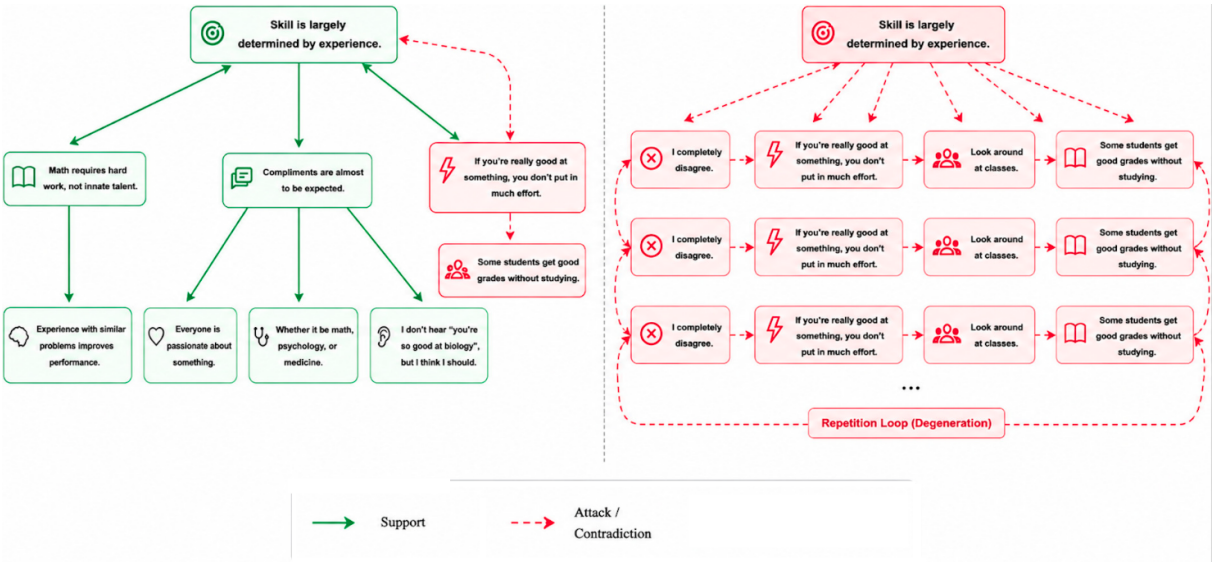


Figure 7: Qwen collapse on CMV. The ground-truth (left), i.e., human-annotated data structure exhibits coherent argumentative reasoning, combining supporting and attacking relations with diverse evidence and contextual examples. In contrast, the Qwen output (right) collapses into repetitive generation patterns characterized by cyclic reuse of a small set of statements, structural degeneration, and the appearance of unsupported nodes not present in the reference graph. The generated structure loses argumentative diversity and progressively converges toward repeated motifs rather than expanding the discourse.

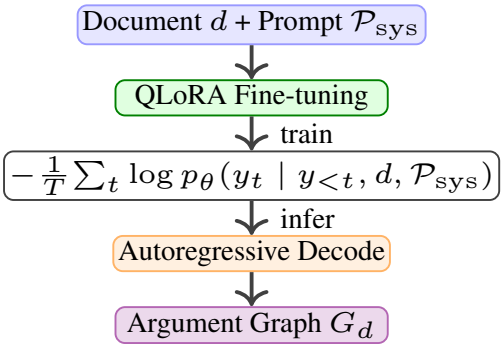


Figure 11: SFT: QLoRA adapters trained via causal LM loss over gold completion tokens.

B.4 Cold-Start Reinforcement Fine-Tuning (RFT)

We employ GRPO algorithm (Shao et al., 2024). For each input query q, the model generates N candidate argument graphs $\{o_1, \ldots, o_N\}$. Each candidate o_i is scored using a rule-based reward function r_i, and the policy is updated to favor outputs yielding higher rewards. Formally, we define the importance ratio and the group-normalized advantage as

$$\rho_i = \frac{\pi_\theta(o_i \mid q)}{\pi_{\text{ref}}(o_i \mid q)}, \quad \hat{A}_i = \frac{r_i - \bar{r}}{\sigma(r)},$$

where:

- $\bar{r} = \frac{1}{N}\sum_{j=1}^{N} r_j$ and $\sigma(r)$ is the standard deviation of rewards across the N candidates.
- π_θ: current policy parameterized by θ
- π_{ref}: reference policy (base LLM)

The full GRPO objective is then:

$$\mathcal{L} = -\frac{1}{G}\sum_{i=1}^{G} \hat{A}_i \cdot \min\Big(\rho_i, \text{clip}(\rho_i, 1-\varepsilon, 1+\varepsilon)\Big) + \beta\, \text{KL}(\pi_\theta \,||\, \pi_{\text{ref}}), \tag{5}$$

where G is the group size (number of sampled candidates per input), ε is the clipping threshold that constrains the importance ratio ρ_i, β is the KL-penalty coefficient, and the KL-divergence term $\text{KL}(\pi_\theta \,||\, \pi_{\text{ref}})$ constrains the updated policy π_θ from drifting too far from the reference policy π_{ref}, ensuring that the model preserves prior knowledge while still improving with respect to the reward function.

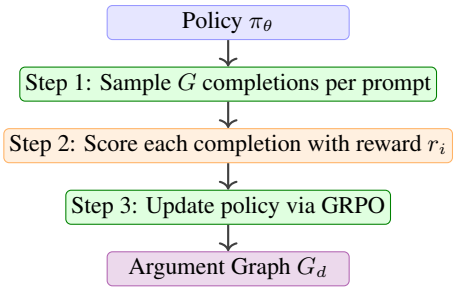


Figure 12: RFT: Cold-start reinforcement fine-tuning using GRPO.

B.5 Evaluation Metrics

B.5.1 Node F1 via Word-Jaccard Matching

Since predicted spans may not align exactly with gold spans, we use word-level Jaccard similarity rather than strict string matching. Strict matching would penalise outputs that are semantically equivalent to gold spans but differ in minor surface form, for instance when a generative model corrects punctuation, normalises whitespace, or slightly adjusts tokenisation boundaries while preserving the exact argumentative content. Word-level Jaccard tolerates such variations by measuring token-set overlap, rewarding predictions that recover the correct content words regardless of incidental surface differences. Let $W(\cdot)$ denote the set of word tokens of a text span. For a predicted span $\hat{s}$ and a gold span s, the word Jaccard similarity is:

$$J(\hat{s},\ s) = \frac{\left|W(\hat{s}) \cap W(s)\right|}{\left|W(\hat{s}) \cup W(s)\right|}$$

We construct a similarity matrix $\mathbf{S} \in [0,1]^{|\hat{V}|\times|V|}$ over all predicted–gold span pairs, zero out entries below threshold $\theta = 0.5$, then retain at most one entry per row and column (one-to-one matching), yielding binary alignment matrix $\mathbf{A} \in \{0,1\}^{|\hat{V}|\times|V|}$.

Let $\mathrm{TP}_N = \sum_{i,j} A_{ij}$ be the number of matched node pairs. Node-level precision, recall, and F1 are then:

$$P_N\ (\text{Node Precision}) = \frac{\mathrm{TP}_N}{|\hat{V}|} \tag{6}$$

$$R_N\ (\text{Node Recall}) = \frac{\mathrm{TP}_N}{|V|} \tag{7}$$

$$\text{Node F1} = \frac{2\,P_N\,R_N}{P_N + R_N} \tag{8}$$

B.5.2 Edge F1 via Exact Triplet Matching

Each predicted edge is a triplet $(\hat{s}_{\text{src}},\ \hat{s}_{\text{tgt}},\ \hat{\lambda})$: source span, target span, and relation label. We construct a binary matching matrix $\mathbf{T} \in \{0,1\}^{|\hat{E}|\times|E|}$ where entry $T_{kl} = 1$ iff *all three* conditions hold:

- $J(\hat{s}_{\text{src},k},\ s_{\text{src},l}) \geq \theta$ (source spans match)
- $J(\hat{s}_{\text{tgt},k},\ s_{\text{tgt},l}) \geq \theta$ (target spans match)
- $\hat{\lambda}_k = \lambda_l$ (relation label agrees exactly)

Applying one-to-one matching to $\mathbf{T}$ results in alignment matrix $\mathbf{B}$. Let $\mathrm{TP}_E = \sum_{i,j} B_{ij}$. Edge-level precision, recall, and F1 are:

$$P_E\ (\text{Edge Precision}) = \frac{\mathrm{TP}_E}{|\hat{E}|} \tag{9}$$

$$R_E\ (\text{Edge Recall}) = \frac{\mathrm{TP}_E}{|E|} \tag{10}$$

$$\text{Edge F1} = \frac{2\,P_E\,R_E}{P_E + R_E} \tag{11}$$

Edge matching is strictly harder than node matching: both endpoint spans must satisfy $J \geq \theta$ *and* the relation label must agree exactly.

B.6 Training Hyperparameters

All experiments were conducted on single NVIDIA GPUs. Most runs used an A100 80GB. Models were loaded in 4-bit NF4 quantization (QLoRA) using the Unsloth library for accelerated fine-tuning.

LoRA configuration (shared across SFT and RFT). Rank r=16, α=32 (scaling α/r=2.0), applied to all linear projections (`q`, `k`, `v`, `o`, `gate`, `up`, `down`). Gradient checkpointing uses the Unsloth-optimized variant. Maximum prompt length is 4 096 tokens; maximum completion length is 2 048 tokens.

SFT Training uses `trl.SFTTrainer`

Parameter	Value
Learning rate	5×10^{-5}
Optimizer	AdamW 8-bit (β_1=0.9, β_2=0.999)
LR scheduler	Cosine with 10% linear warmup
Epochs	15

Table 9: SFT hyperparameters.

RFT Training uses `trl.GRPOTrainer`.

C Experimental Results

Parameter	Value
Learning rate	1×10^{-5}
KL penalty (β)	0.1
Optimizer	AdamW 8-bit (β_1=0.9, β_2=0.999)
Num. generations	8 per prompt
Epochs	15

Table 10: RFT (GRPO) hyperparameters.

Train → Test	Model	Node F1	Edge F1	F1F1
AAE → AbstRCT	Llama	0.602	0.077	0.101
	Mistral	0.609	0.096	0.123
	Qwen	0.224	0.025	0.035
AAE → CMV	Llama	0.527	0.065	0.098
	Mistral	0.418	0.061	0.098
	Qwen	0.041	0.000	0.000
AbstRCT → AAE	Llama	0.618	0.067	0.102
	Mistral	0.646	0.157	0.218
	Qwen	0.384	0.081	0.116
AbstRCT → CMV	Llama	0.305	0.017	0.024
	Mistral	0.341	0.018	0.027
	Qwen	0.020	0.000	0.000
CMV → AAE	Llama	0.769	0.166	0.235
	Mistral	0.606	0.098	0.140
	Qwen	0.307	0.046	0.069
CMV → AbstRCT	Llama	0.588	0.022	0.032
	Mistral	0.391	0.009	0.013
	Qwen	0.218	0.016	0.023

Table 11: SFT cross-domain results across all train-test pairs. Edge F1 collapses near zero in every configuration regardless of model size or training corpus.

Train → Test	Model	Node F1	Edge F1	F1F1
AAE → AbstRCT	Llama	0.510	0.040	0.080
	Mistral	0.568	0.073	0.105
	Qwen	0.294	0.065	0.083
AAE → CMV	Llama	0.522	0.052	0.085
	Mistral	0.402	0.009	0.016
	Qwen	0.093	0.017	0.026
AbstRCT → AAE	Llama	0.767	0.110	0.176
	Mistral	0.678	0.082	0.132
	Qwen	0.308	0.057	0.083
AbstRCT → CMV	Llama	0.570	0.026	0.045
	Mistral	0.449	0.010	0.017
	Qwen	0.132	0.007	0.011
CMV → AAE	Llama	0.783	0.101	0.164
	Mistral	0.536	0.052	0.083
	Qwen	0.199	0.033	0.050
CMV → AbstRCT	Llama	0.616	0.067	0.103
	Mistral	0.523	0.054	0.080
	Qwen	0.175	0.028	0.040

Table 12: RFT cross-domain results across all train-test pairs. Edge F1 remains near zero under all domain shifts.