Explainable End-to-End Argument Mining with Toulmin-Structured Explanations

Zhiqiang Shi
Department of Informatics
King's College London
zhiqiang.shi@kcl.ac.uk

Oana Cocarascu
Department of Informatics
King's College London
oana.cocarascu@kcl.ac.uk

Abstract

Large language models (LLMs) have advanced argument mining (AM), yet most approaches treat component classification, relation identification, and explanation generation as separate tasks, limiting coherence and explainability. We propose the first explainable end-to-end AM framework grounded in Toulmin's theory, jointly predicting argument components, argumentative relations, and structured explanations in a single generation step. This unified formulation encourages LLMs to produce internally consistent argument graphs and reasoning traces aligned with their predictions. Further, we introduce five explainability criteria tailored to AM and grounded in Toulmin's theory: completeness, correctness, faithfulness, soundness, usefulness, and develop a unified evaluation protocol combining structural metrics, neural similarity, LLM-based judgment, human evaluation and qualitative analysis. Extensive evaluation on several LLMs shows that our approach achieves competitive performance while substantially improving explainability, generating Toulmin-based explanations that are complete, grounded and faithful, uncovering reasoning errors not captured by standard AM metrics, and being consistently preferred in usefulness evaluations. [1]

1 Introduction

Argument mining (AM) aims to extract argumentative structures from text (Lawrence and Reed, 2019), aiding systems understand the reasoning behind opinions or conclusions and supporting tasks such as scientific evidence synthesis, legal reasoning (Palau and Moens, 2009), policy making (Sardianos et al., 2015). Recently, large language models (LLMs) have shown strong general-purpose reasoning and generation capabilities, leading to growing interest in applying them to AM (Al Zubaer et al., 2023; Cabessa et al., 2025). However, most approaches treat the core subtasks (argument component classification, argument relation identification, argument relation classification) separately, overlooking how arguments naturally function as coherent structures and limiting the ability of explainability methods to capture the full reasoning chain that links components to relations. Additionally, common approaches such as chain-of-thought (Wei et al., 2022) are unstructured and cannot guarantee that argumentative elements (supporting premises, attacks) are faithfully represented.

To address these limitations, we propose an explainable end-to-end AM framework that jointly predicts argumentative components, their relations, and Toulmin-structured explanations in a single generation pass. We draw on Toulmin's theory of argumentation (Toulmin, 1958) (claim, data, warrant, rebuttal) to provide a structured framework that encourages models to produce coherent and faithful reasoning aligned with their predicted argument graphs. This enables us to examine not only whether LLMs identify the correct argumentative structure, but also how they justify it.

Furthermore, given the graph-structured outputs of AM (e.g. components, relations), we introduce five explainability criteria tailored to AM, completeness, correctness, faithfulness, soundness and usefulness. We evaluate several LLMs under a unified protocol that combines automated structural metrics, neural evaluation methods, LLM-based scoring and qualitative analysis, addressing a key gap in explainability and evaluation for AM.

Our contributions are as follows: **1)** We introduce the first explainable end-to-end argument mining framework that jointly predicts components, relations, and Toulmin-structured explanations within a unified generation process; **2)** We propose five explainability criteria, completeness, correctness, faithfulness, soundness and usefulness, tailored to argument mining and grounded in argu-

[1] Code can be found at: https://github.com/zqshizhiqiang/toulmin_argument_mining/

Proceedings of the 15th International Joint Conference on Natural Language Processing and the 5th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 3575–3591
November 6-10, 2026 ©2026 Association for Computational Linguistics

mentation theory; **3)** We develop a comprehensive evaluation protocol combining structural metrics, neural similarity measures, LLM-based judgments, human evaluation and qualitative analysis.

2 Related Work

Argument Mining. Early AM works relied on supervised learning where task-specific models were trained from scratch (Stab and Gurevych, 2014; Niculae et al., 2017; Bouslama et al., 2019) and later on pre-trained language models, (Srivastava et al., 2022; Liu et al., 2023). LLMs have further advanced AM, with in-context learning (Al Zubaer et al., 2023; Chen et al., 2024; Abkenar et al., 2024; Otiefy and Alhamzeh, 2024; Gorur et al., 2025) and supervised fine-tuning (Cabessa et al., 2025) achieving strong results and enabling generative formulations of the AM pipeline. Whilst end-to-end AM has been extensively studied (Eger et al., 2017; Persing and Ng, 2016; Morio et al., 2022; Bao et al., 2022), most LLM-based works (Al Zubaer et al., 2023; Chen et al., 2024; Gorur et al., 2025; Abkenar et al., 2024; Cabessa et al., 2025) still treat sub-tasks separately. Only limited works e.g. Savigny and Yun (2025) explore multitask learning with LLMs, without addressing end-to-end AM. This limitation, combined with the lack of explainability, limits their practical use.

Explainability in NLP. Recent surveys (Lyu et al., 2024; Mohammadi et al., 2025) highlight that neural models often produce unfaithful explanations that do not accurately reflect their internal reasoning. Chain-of-thought (Wei et al., 2022) is commonly used to provide explanations for the model's predictions, but recent works (Lyu et al., 2023; Tanneru et al., 2024) show that these free-text explanations often contain latent inconsistencies, hallucinated content, and missing inferential steps.

The Toulmin (1958) model of argumentation is widely used in argumentation studies. Guided by this theory, Gupta et al. (2024) propose the argument explication task that uses LLMs to make the argument's structure and implicit reasoning explicit. However, the method requires multiple forward passes per argument, limiting its applicability to large-scale inputs (e.g. medical abstracts, essays).

Our work situates itself within the explainability landscape by integrating Toulmin-grounded explanations directly into the end-to-end AM process. Unlike prior studies that apply Toulmin analysis post-hoc (Dalzell et al., 2026) or require multi-pass prompting to extract reasoning components (Gupta et al., 2024), we generate components, relations, and structured Toulmin explanations simultaneously in a single pass, producing a unified and intrinsically interpretable AM pipeline.

End-to-end AM with Toulmin explanations

Input: Biomedical abstract with `<ACi>` components.
Task (single LLM pass):

1. Classify each `ACi` as **Claim** or **Premise**.
2. Identify **support** or **attack** relations.
3. For each Claim, output a **Toulmin structure**: data, warrant, rebuttal.

Output: One JSON object with:
`component_types`, `argument_relation_types_list`, `toulmin_structures`.

Figure 1: Method overview: unified prompt for component typing, relation prediction, and Toulmin explanation generation.

3 Methodology

In this work, we propose an explainable end-to-end AM pipeline grounded by Toulmin's theory. In the following, we introduce our proposed framework and the different criteria used to evaluate the explainability of the generated Toulmin structures.

3.1 Task formulation

We formulate end-to-end AM as a structured generation task that jointly predicts argument component types, argumentative relations, and Toulmin-style explanations from an input paragraph. [2] Figure 1 illustrates our proposed method. Let an input paragraph contain n argument components, each annotated with span-level tags $\{AC_1, AC_2, \ldots, AC_n\}$. Each component AC_i consists of a text span and an unknown label to be inferred by the model.

Argument Component Classification (ACC). The first subtask is to assign each argument component a type from a given set, for example $\mathcal{T} = \{\text{Claim}, \text{Premise}\}$. The output of this step is a mapping $f_{\text{type}} : AC_i \mapsto \mathcal{T}$.

Argument Relation Identification and Classification (ARIC). The second subtask is to identify directed argumentative relations and the relation types between component pairs. For each ordered pair (AC_i, AC_j), the model predicts whether a relation exists and, if so, assigns it a type from a given

[2] In our work, we use *end-to-end* to refer to jointly predicting argument components, argumentative relations, and structured explanations in a single generation step given the segmented text rather than the raw text. Following Liu et al. (2023); Cabessa et al. (2025), we directly use the segmented text as input and assume segmentation is solved.

set (e.g. $\mathcal{R} = \{\text{support}, \text{attack}\}$). This yields a set of typed edges $E = \{(AC_i, AC_j, r_{ij}) \mid r_{ij} \in \mathcal{R}\}$.

Toulmin Structure Construction. For each predicted Claim component AC_k, the model constructs a structured explanation following Toulmin's argumentation model. Given all premises that support or attack AC_k, the model produces: *data* (the set of supporting premises), *warrant* (a natural-language reasoning statement linking the data to the claim), *rebuttal* (any premises that attack the claim). Formally, for each claim AC_k, we define the Toulmin structure $\tau(AC_k) = \big(\text{data}_k, \text{warrant}_k, \text{rebuttal}_k\big)$ where $\text{data}_k, \text{rebuttal}_k \subseteq \{AC_1, \ldots, AC_n\}$.

Unified Generation Objective. Unlike methods that address subtasks independently, we frame AM as a single sequence-to-structure generation problem. Given a paragraph annotated with component spans, the model generates a unified JSON object $\mathcal{Y} = \{f_{\text{type}}, E, \tau\}$ containing component types, relations, and Toulmin structures in a single forward pass. This formulation allows the model to jointly reason over the global argument structure and generate explanations grounded in Toulmin's theory.

3.2 Toulmin-Grounded Explainability Criteria

Our goal is to generate interpretable argumentative structures that align with both the explicit component-relation graph and the implicit reasoning behind argumentative links. We introduce five explainability criteria tailored to AM and grounded in Toulmin's theory that define the desired properties of the generated explanations and guide our evaluation protocol.

Completeness measures whether the explanation covers all argument components and relations referenced in the model's output. A Toulmin structure is complete if every supporting or attacking component predicted by the model is reflected in the *data* or *rebuttal* fields, and if the explanation accounts for all relevant links in the argumentative structure.

Correctness assesses the alignment between the explanation and the gold-standard argument structure. An explanation is correct if the premises and attacking components listed in the Toulmin structure correspond to their true argumentative roles.

Faithfulness captures the degree to which the explanation reflects the model's own reasoning process rather than a post-hoc rationalization. In our setting, a faithful explanation exhibits consistency between the model's predicted component types, relation predictions, and the reasoning expressed in the Toulmin structures, forming a coherent interpretation of the underlying argumentative graph.

Soundness evaluates two dimensions: factual grounding and the logical quality of the explanation. An explanation is considered sound only when it is both factually grounded and logically valid.

Groundedness: The explanation must rely solely on information present in the input paragraph. Data and warrants that introduce hallucinated facts or unsupported claims are considered ungrounded and therefore unsound.

Logical validity: Following Toulmin's model, a warrant is logically valid when it provides a correct, coherent inferential link from Data to Claim. A valid warrant expresses an appropriate generalization or rule, avoids logical fallacies, and maintains domain-appropriate reasoning.

Usefulness measures the explanation's interpretability to human readers on three dimensions.

Practical usefulness: how effectively the explanation helps users understand the predicted argument components and relations.

Actionability: how well the explanation enables users to verify, critique, or further analyze the model's predictions.

Trust support: how well the explanation helps users assess the reliability of the prediction without introducing misleading or hallucinated content.

Our explainability criteria builds on argumentation theory and explainable NLP. Logical validity follows Toulmin's model of warrants (Toulmin, 1958), while faithfulness and groundedness follow principles established in interpretable NLP (Jacovi and Goldberg, 2020; Maynez et al., 2020; Lyu et al., 2024) and relates to factuality and hallucination studies in natural language generation (Maynez et al., 2020). Our definitions of completeness and correctness draw insights from (Danilevsky et al., 2020; Lyu et al., 2024), and usefulness connects to human-centered interpretability research (Zhang et al., 2025; Al-Ansari et al., 2024).

Together, these criteria provide a comprehensive conceptual foundation for evaluating the interpretability and quality of model-generated Toulmin structures. They also support a hybrid evaluation framework that integrates automated metrics, neural similarity scores, LLM-based judgments, human evaluation, and qualitative analysis.

4 Experiment Setup

Datasets. We evaluate our method on two AM datasets. *AbstRCT* (Mayer et al., 2020) is a biomedical dataset containing 650 scientific abstracts annotated with argumentative components and relations. Following Liu et al. (2023), we merge MajorClaim and Claim into a single category, and we use the *neo* test split as adopted by Morio et al. (2022). *CDCP* (Park and Cardie, 2018) comprises 731 user-generated comments from online policy discussion platforms, annotated for argument structure. We conduct all experiments on the test set using prompting. The experiment results for ACC and ARIC are measured using macro F_1 score. Following Cabessa et al. (2025), the macro F_1 score is computed over the whole test set.

Models. We use Gemma 3 27B, LLaMA 3.3 70B and Qwen 3 235B A22B Instruct 2507 (referred to as Qwen 3 235B throughout the paper), accessed using public APIs, we use greedy decoding for all the experiments (temperature is 0.0, `top_p` is 1.0).[3]

Baselines. For ACC, we adopt the component-wise prompting setup (Al Zubaer et al., 2023; Chen et al., 2024) where each argument component is provided to the LLM independently for type classification. For ARIC, we follow Gorur et al. (2025) and cast relation prediction as a pairwise classification task: for each paragraph, we enumerate all possible ordered component pairs and query the LLM individually to output the appropriate relation type or `none` if no argumentative relation exists (**baseline**). We also include an internal baseline in which we use a similar prompt as our Toulmin framework but the LLM is prompted to output only the AM predictions without Toulmin explanations (**end-to-end**), and a chain-of-thought (Wei et al., 2022) baseline where the LLM is asked to output its reasoning steps before giving the AM predictions (**CoT**).

5 Argument Mining Task Performance

Table 1 compares different prompting strategies on AbstRCT and CDCP. The end-to-end formulation significantly outperforms the component-wise (ACC) and pairwise (ARIC) baselines, indicating that LLMs can benefit from the additional context. The Toulmin prompt is competitive or superior to the prediction-only end-to-end method. In most settings, the CoT baseline performs worse than both the Toulmin prompt and the baseline, highlighting that free-form reasoning does not reliably guide LLMs toward correct argumentative structure. Overall, these results show that the Toulmin prompt provides competitive or improved performance compared to the end-to-end baseline while providing a degree of explainability.

[3]See Appendix C for prompting configurations.

Model	AbstRCT		CDCP	
	ACC	ARIC	ACC	ARIC
Best non-LLM results	0.928	0.750	0.825	-
Gemma3 27B baseline	0.758	0.311	0.348	0.187
Gemma3 27B end-to-end	**0.798**	**0.493**	0.462	0.378
Gemma3 27B Toulmin	0.751	**0.493**	**0.464**	0.393
Gemma3 27B CoT	0.722	0.479	0.416	**0.404**
LLaMA3.3 70B baseline	0.414	0.322	0.466	0.281
LLaMA3.3 70B end-to-end	0.707	**0.539**	0.495	0.374
LLaMA3.3 70B Toulmin	**0.739**	0.531	**0.509**	**0.387**
LLaMA3.3 70B CoT	0.737	0.538	0.432	0.364
Qwen3 235B end-to-end	**0.878**	0.570	**0.503**	0.417
Qwen3 235B Toulmin	0.855	**0.574**	0.499	**0.418**

Table 1: Macro F_1 scores for different prompting strategies. Best non-LLM results are from Liu et al. (2023) for AbstRCT, and Bao et al. (2021) for CDCP. As the baseline method requires multiple LLM calls, and the CoT method produces long outputs increasing the cost, we only run these experiments on some of the models.

6 Explainability Analysis

Next, we systematically analyze the additional explainability enabled by Toulmin structures (see Section 3.2 for explainability criteria).

6.1 Completeness

Completeness assesses whether the model's explanations include all argument components and relations referenced in its predictions. To measure this, we use two coverage metrics: `data_cov` which measures the percentage of predicted support relations for a claim that are included in the claim's data list, and `rebuttal_cov` which measures the percentage of predicted attack relations for a claim that are included in the claim's rebuttal list. Table 2a shows that `data_cov` and `rebuttal_cov` are high for all models, indicating the high completeness of Toulmin explanations.

6.2 Correctness

Correctness evaluates whether the explanatory elements generated by the model accurately reflect the gold argumentative structure. We assess correctness using five metrics. `DataCorrect` computes the proportion of premises listed in the Data field that correspond to true supporting components in the gold graph, while `RebutCorrect` mea-

Model	data_cov	rebuttal_cov
Gemma 3 27B	0.944	0.870
LLaMA 3.3 70B	0.919	0.989
Qwen 3 235B	0.992	0.931

(a) Completeness evaluation.

DataCorrect	RebutCorrect	MissErr	FP_Err	OntoCorrect
0.585	0.179	0.268	0.367	0.691
0.643	0.232	0.154	0.369	0.736
0.720	0.194	0.105	0.283	0.752

(b) Correctness metrics for Toulmin explanations (macro-averaged).

Table 2: Toulmin explanation evaluation: (a) completeness and (b) correctness.

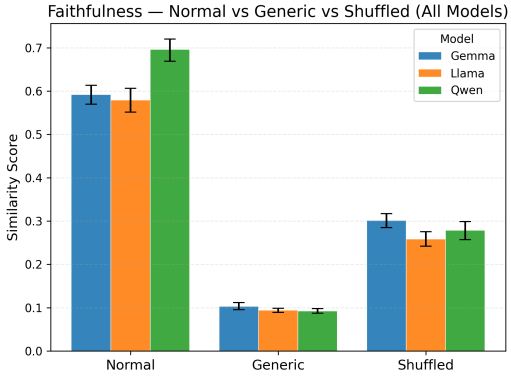


Figure 2: Semantic faithfulness of warrant.

sures the proportion of listed rebuttals that match gold attacks. To capture missing or spurious evidence, we report `MissErr` (the fraction of gold supports or attacks omitted from the explanation) and `FP_Err` (the fraction of included supports or attacks that do not appear in the gold structure). Finally, `OntoCorrect` assesses whether the edges implied by the Toulmin structure are consistent with the gold graph ontology (i.e. whether the nodes of the edge appear in the gold graph, and whether the direction follows the gold graph ontology, e.g. premise $\rightarrow$ claim rather than claim $\rightarrow$ premise).

Table 2b shows that the overall correctness is high across different models (high `DataCorrect`, `OntoCorrect`, and low `MissErr`, `FP_Err`), and larger models give better results. However, `RebutCorrect` is low for all models, reflecting the sparsity and difficulty of attack relations. Overall, the correctness results show that while current LLMs can capture core supporting evidence with moderate reliability, generating fully accurate and ontology-consistent Toulmin explanations, particularly for rebuttals, remains challenging.

6.3 Faithfulness

We follow prior definitions of faithfulness as internal consistency between an explanation and the model's reasoning process behind its predictions (Jacovi and Goldberg, 2020; Lyu et al., 2024). In this section, we tailor faithfulness evaluation to AM and evaluate it from three dimensions.

Model	Support F1	Attack F1	Combined
Gemma 3 27B	0.686	0.833	0.760
LLaMA 3.3 70B	0.809	0.705	0.757
Qwen 3 235B	0.964	0.725	0.845

Table 3: Structural faithfulness: macro F_1 between predicted edges and edges implied by the model's Toulmin explanations (*Data* $\Rightarrow$ support, *Rebuttal* $\Rightarrow$ attack).

6.3.1 Structural Faithfulness

We measure structural faithfulness as the internal consistency between the model's predicted argumentative relations and the edges implied by its Toulmin explanations. Concretely, for each document d, let $S_d^{\text{pred}} = \{(i \rightarrow k)\}$ and $A_d^{\text{pred}} = \{(j \rightarrow k)\}$ denote the sets of predicted support and attack edges from the model's relation list, respectively. From the Toulmin outputs, each entry in a claim's Data/Rebuttal field implies a support/attack edge to that claim, denoted as $S_d^{\text{expl}} = \{(i \rightarrow k)\}$ and $A_d^{\text{expl}} = \{(j \rightarrow k)\}$. We compute per-document precision, recall, and F_1 for supports by comparing S_d^{pred} with S_d^{expl}, and similarly for attacks by comparing A_d^{pred} with A_d^{expl}. The Support $F1$ and Attack $F1$ are macro-averaged over documents We also report a Combined score, calculated as the mean of Support F_1 and Attack F_1 to capture the overall structural consistency.

Table 3 show that all models maintain a substantial degree of internal consistency between their predicted relation graphs and the edges implied by their Toulmin explanations. Overall, Qwen 3 235B obtains the highest combined faithfulness score (0.845), demonstrating that larger models produce Toulmin explanations that more faithfully reflect their own structural predictions.[4]

6.3.2 Semantic Faithfulness

We also measure the semantic faithfulness of the model's generated Toulmin explanations. As the

[4]This metric evaluates *internal* consistency between the relation list and the explanation, distinct from correctness w.r.t. the gold graph.

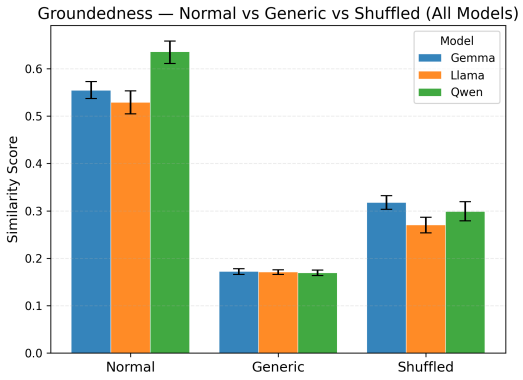


Figure 3: Groundedness of warrant.

Data and Rebuttal fields contain AC tags that directly reference the input text, with only Warrant being free text, we assess faithfulness primarily through the warrant. A faithful warrant should reflect the support relations[5] indicated by the model's predictions without introducing any hallucinated content. To measure this, we first textualize each predicted support relation as `source_AC provides support evidence to target_AC`, replacing AC tags with their actual spans. We then concatenate all such textualized relations as input and compute its similarity to the warrant using a sentence-transformer model.[6]

To contextualize these scores, we include two baselines: *Generic warrant*, replacing all warrants with a fixed template ("When the available evidence is relevant and sufficient, it supports the claim."), and *Shuffled warrant*, randomly reassigning warrants across the test set so that each claim is paired with an unrelated warrant. We compute similarity scores for all warrants and estimate uncertainty using bootstrap resampling (Efron and Tibshirani, 1986) with 5000 samples; Figure 2 shows the mean and 95% confidence intervals. Across all models, normal warrants achieve substantially higher similarity scores than both baselines, indicating strong semantic faithfulness with respect to the model's predicted support relations. Similarly to the structural faithfulness, Qwen 3 235B obtains the highest score, demonstrating that the faithfulness of the generated explanations relies on LLM's capability.

[5]Here, we only need to consider support relations because in Toulmin's theory, warrant establishes links between data (not rebuttals) and claim.

[6]https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

Model	CausalAcc	PredUnstability
Gemma 3 27B	0.848	0.221
LLaMA 3.3 70B	0.885	0.151
Qwen 3 235B	0.865	0.127

Table 4: The perturbation results for different models.

6.3.3 Perturbation Tests

To further measure the faithfulness of Toulmin-style explanations generated by the model, we perform several perturbation tests. For each document, we extract a canonical prediction graph $\mathcal{E}^0_{\text{pred}}$ from the model predicted *argument relation list*. We also derive an explanation edge set $\mathcal{E}^0_T$ from the *Toulmin structure* (data⇒support, rebuttal⇒attack). For every $(a \to C, r) \in \mathcal{E}^0_T$ that also appears in $\mathcal{E}^0_{\text{pred}}$, we neutralize/delete the text of AC a and re-run the model. For deletion, we replace the AC text with a fixed tag `[REMOVED]`, for neutralization, we replace the AC text with a neutral sentence `This sentence provides background information without evidence`. A faithful explanation implies that the predicted relation $(a \to C, r)$ should disappear if AC a is deleted or neutralized. Let y be the predicted reaction (whether disappeared or not) and $\widehat{y}$ the expected one; we report **CausalAcc** $= \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\{\, y_i = \widehat{y}_i \,\}$ over all eligible targets.

To make the `CausalAcc` score more meaningful, we add a non-AC baseline in which we perturb background text outside all AC spans and measure how much the model's prediction graph changes. Let $\mathcal{E}^0_{\text{pred}}$ be the canonical prediction graph extracted from the model's predicted argument relations before perturbation, and let $\mathcal{E}'_{\text{pred}}$ be the prediction graph after a non-AC edit. We report **PredUnstability** $= 1 - \frac{|\{\, e \in \mathcal{E}^0_{\text{pred}} : e \text{ is preserved in } \mathcal{E}'_{\text{pred}} \}|}{|\mathcal{E}^0_{\text{pred}}|}$, i.e., the fraction of canonical prediction edges whose existence or polarity changes under the non-AC edit. This establishes that observed changes for `CausalAcc` are not due to generic sensitivity but are tied to explanation-guided AC edits.

The results are shown in Table 4. For all models, we observe strong explanation-to-prediction causal responsiveness (high `CausalAcc` score), indicating that when we perturb an AC cited in the model's Toulmin explanation, the model's predicted argument relation list changes in the expected direction in a high fraction of eligible cases. For the non-AC baseline, the relatively low `PredUnstability` shows that only a small fraction of canonical prediction edges change

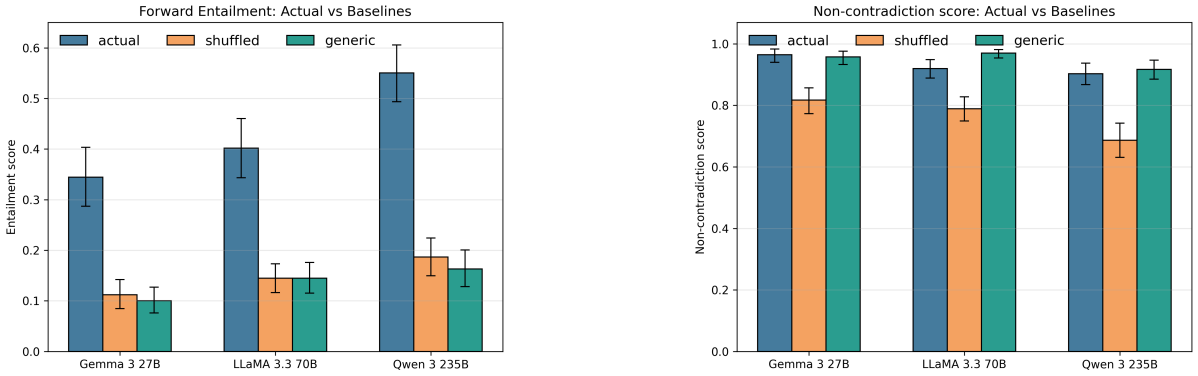


Figure 4: Forward entailment and non-contradiction score across different models.

Evaluator	Wins (Toulmin/CoT)	Mean Δ
Annotator 1	37 / 12	+3.04
Annotator 2	37 / 10	+2.06
Qwen 3 235B	40 / 10	+3.26
DeepSeek V3.2	48 / 2	+7.96
GPT 4.1	48 / 2	+5.46

Table 5: Usefulness comparison between Toulmin and CoT explanations. Mean Δ is computed as the average total score across all examples of Toulmin explanation minus that of CoT.

under non-AC (background-only) edits, confirming that most prediction changes are driven by explanation-guided AC edits rather than incidental perturbations. These results further suggest the faithfulness of Toulmin explanations.

6.4 Soundness

6.4.1 Groundedness

For groundedness, we use the similar method as for semantic faithfulness in Section 6.3.2. As the data and rebuttal fields are AC tags that refer to the input text, the groundedness of the Toulmin explanations depends on the groundedness of the warrant. For a grounded warrant, it must solely rely on information present in the input paragraph without introducing hallucinated content. Therefore, we use a sentence transformer model [7] to compute the cosine similarity between each warrant and the input paragraph. To make the similarity score more meaningful, we also include the same baselines as in Section 6.3.2. We compute the groundedness score for all warrants; Figure 3 shows the mean and 95% confidence intervals. We observe that all models produce warrants with substantially higher similarity scores than both baselines. This indicates that the generated warrants reliably draw upon information present in the paragraph, demonstrating strong groundedness of the Toulmin-structured explanations.

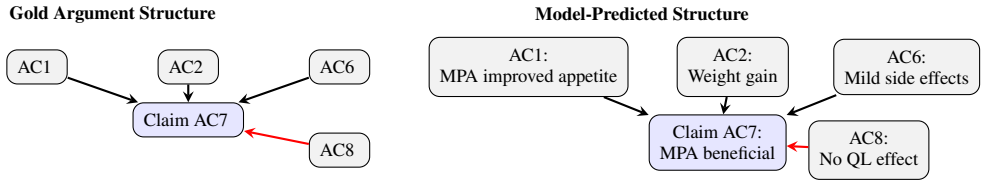
6.4.2 Logical Validity

Logical validity measures whether the warrant can provide a correct and coherent inferential link from data to claim (see Section 3.2). We evaluate this using a natural language inference (NLI) formulation with two scores: *Forward entailment* where we concatenate the data and warrant as the premise and use the claim as the hypothesis, computing the entailment score with an NLI model,[8] and *Non-contradiction* where we concatenate the claim and data as the premise and treat the warrant as the hypothesis, using the contradiction probability ($contradiction_score$) to compute a non-contradiction score of $1 - contradiction_score$. Low non-contradiction indicates that the warrant conflicts with the claim or data and is therefore logically invalid. As in Section 6.3.2, we include generic-warrant and shuffled-warrant baselines for comparison.

Figure 4 shows that all models achieve substantially higher forward-entailment scores than both baselines, indicating a strong logical connection between data, warrant, and claim. For non-contradiction, the generic warrant provides an upper bound, as it should not conflict with any claim or data. The generated warrants match this upper-bound behavior, while randomly shuffled warrants exhibit a clear drop in non-contradiction. These results indicates the logical validity of the generated warrant in the Toulmin explanations.

[7] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

[8] https://huggingface.co/FacebookAI/roberta-large-mnli



Toulmin Explanation for Claim AC7

Data: AC2 (weight gain), AC6 (mild toxicity), AC1 (appetite improvement).
Warrant: *"Improved appetite, weight stabilization, and a favorable side-effect profile together justify that MPA is clinically beneficial."*
Rebuttal: AC8 (no measurable improvement in overall QL).
Text snippets shortened for clarity.

Figure 5: Example illustrating how the Toulmin scaffold surfaces the model's reasoning. Supporting ACs appear as **data**, the predicted attack appears as a **rebuttal**, and the **warrant** provides the missing causal explanation.

6.5 Usefulness

To assess the usefulness of Toulmin explanations, we conduct both human evaluation and LLM-as-a-judge comparisons (Gu et al., 2024) between Toulmin and CoT explanations across the three dimensions of practical usefulness, actionability, and trust support (Section 3.2). Annotators and LLM judges[9] are shown the predictions and corresponding explanations generated by the Toulmin and CoT prompts[10] and rate each explanation on a 1–5 scale for each dimension before selecting an overall winner. For this evaluation, we generate explanations using Gemma3 27B and use Qwen3 235B, DeepSeek V3.2, and GPT-4.1 as judges.

Table 5 shows that Toulmin explanations are preferred over CoT by large margins, both in win rate and average usefulness scores.[11] We also compute the raw agreement and Cohen's κ: the three judges exhibit moderate raw agreement (0.66–0.86) and fair beyond-chance agreement (0.296-0.479). These results indicate that, while the evaluation contains inherent subjectivity (as expected for explanation usefulness judgments), the relative preference of explanation types is stable across judge models. The human evaluation results follow a similar trend, further strengthening our conclusions.

6.6 Qualitative Analysis

To illustrate how Toulmin-structured generation improves interpretability, we qualitatively examined several AbstRCT examples, focusing on how the Data, Warrant, and Rebuttal fields complement the model-predicted argument structure. The Toulmin explanation surfaced reasoning signals that are not captured by ACC/ARIC predictions alone. Figure 5 illustrates how the Toulmin explanation surfaces the model's reasoning.

Alignment with predicted relations. The Data field collects all premises predicted as supporters of a given claim, while Rebuttal lists predicted attackers. In Figure 5, Claim AC7 ("MPA is beneficial") is supported by AC1, AC2, and AC6, all of which appear in Data; AC8, predicted as an attacker, appears in Rebuttal. This one-to-one mapping makes structural faithfulness directly inspectable.

Making implicit reasoning explicit. The Warrant expresses the causal link between evidence and claim, e.g., "improved appetite, prevention of weight loss, and a favorable toxicity profile justify that MPA is beneficial." Such reasoning is not recoverable from ACC/ARIC outputs alone, demonstrating how the Toulmin format adds interpretable, domain-plausible justification.

Revealing model errors. Toulmin structures also surface inconsistencies: over-aggregated or misplaced premises that remain hidden under quantitative metrics. Appendix F shows further examples.

Compact, human-interpretable summaries. Overall, the Toulmin format delivers concise, interpretable summaries in which Data, Warrant, and Rebuttal make the model's reasoning clearer than free-form CoT explanations.

7 Conclusions

We proposed an explainable end-to-end AM framework grounded by Toulmin's theory. We showed that, compared to outputting AM predictions only, our method substantially enhances explainability

[9] Appendix A contains details of the human instructions and LLM-judge prompts.
[10] The order of the two explanations is randomized.
[11] Full scores and confidence intervals are in Appendix D Table 9.

while maintaining or slightly improving the performance. Further, we introduce five explainability evaluation criteria tailored to AM and grounded in Toulmin's theory, and conduct a comprehensive explainability evaluation for different LLMs using our proposed Toulmin prompt. The results show that LLMs generate highly complete and largely correct Toulmin structures and demonstrate strong faithfulness. The Toulmin explanations also show strong soundness and superior usefulness compared to CoT explanations. Overall, our findings demonstrate that incorporating Toulmin-grounded explanations meaningfully improves the transparency, diagnostic value, and human-centered explainability of end-to-end AM systems. In future work, we plan to extend our work to different domains.

8 Limitations

There are several limitations to our work. While the method jointly generates components, relations, and explanations in a single pass, its performance is bounded by the inherent limitations of current large language models. As shown in our experiments, larger models exhibit better explainability. Further, although our completeness and correctness metrics show strong coverage for supporting evidence, performance on rebuttals remains consistently lower across models. This reflects both the sparsity of attack relations in existing datasets and the difficulty LLMs face in detecting and justifying counterarguments. Moreover, all experiments were conducted on publicly accessible LLM APIs with fixed decoding strategies. We did not explore fine-tuning, preference optimization, or architectural interventions that could further improve explanation quality or enforce stronger reasoning constraints. As a result, our findings reflect the behavior of general-purpose LLMs rather than specialized AM-optimized systems. Finally, because the LLMs used are trained on broad web-scale corpora, it is possible that widely used argument mining datasets such as AbstRCT, CDCP appear in their pretraining data. This limitation is shared across prior LLM-based AM work. Importantly, all prompting strategies evaluated in this paper use the same underlying pretrained model, meaning any pretraining exposure would affect them equally. Thus, potential leakage cannot explain the relative improvements in explainability or structural coherence introduced by our Toulmin framework.

References

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*.

Noor Al-Ansari, Dena Al-Thani, and Reem S Al-Mansoori. 2024. User-centered evaluation of explainable artificial intelligence (xai): A systematic literature review. *Human Behavior and Emerging Technologies*, 2024(1):4628855.

Abdullah Al Zubaer, Michael Granitzer, and Jelena Mitrović. 2023. Performance analysis of large language models in the domain of legal argument mining. *Frontiers in artificial intelligence*, 6:1278796.

Jianzhu Bao, Chuang Fan, Jipeng Wu, Yixue Dang, Jiachen Du, and Ruifeng Xu. 2021. A neural transition-based model for argumentation mining. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6354–6364, Online. Association for Computational Linguistics.

Jianzhu Bao, Yuhang He, Yang Sun, Bin Liang, Jiachen Du, Bing Qin, Min Yang, and Ruifeng Xu. 2022. A generative model for end-to-end argument mining with reconstructed positional encoding and constrained pointer mechanism. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10437–10449, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Rihab Bouslama, Raouia Ayachi, and Nahla Ben Amor. 2019. Using convolutional neural network in cross-domain argumentation mining framework. In *International Conference on Scalable Uncertainty Management*, pages 355–367. Springer.

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, Abu Dhabi, UAE. Association for Computational Linguistics.

Guizhen Chen, Liying Cheng, Anh Tuan Luu, and Lidong Bing. 2024. 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.

Geordie Dalzell, Chen Liu, Wei Peng, and Xinghuo Yu. 2026. Analysis of large language model prompting and generation using toulmin's model. In *AI 2025: Advances in Artificial Intelligence*, pages 3–14, Singapore. Springer Nature Singapore.

Marina Danilevsky, Kun Qian, Ranit Aharonov, Yannis Katsis, Ban Kawas, and Prithviraj Sen. 2020. A

survey of the state of explainable AI for natural language processing. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 447–459, Suzhou, China. Association for Computational Linguistics.

B. Efron and R. Tibshirani. 1986. Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistical Science*, 1(1):54–75.

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, Vancouver, Canada. Association for Computational Linguistics.

Deniz Gorur, Antonio Rago, and Francesca Toni. 2025. Can large language models perform relation-based argument mining? In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 8518–8534, Abu Dhabi, UAE. Association for Computational Linguistics.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, and 1 others. 2024. A survey on llm-as-a-judge. *The Innovation*.

Ankita Gupta, Ethan Zuckerman, and Brendan O'Connor. 2024. Harnessing toulmin's theory for zero-shot argument explication. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10259–10276, Bangkok, Thailand. Association for Computational Linguistics.

Alon Jacovi and Yoav Goldberg. 2020. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4198–4205, Online. Association for Computational Linguistics.

John Lawrence and Chris Reed. 2019. Argument mining: A survey. *Computational Linguistics*, 45(4):765–818.

Boyang Liu, Viktor Schlegel, Riza Batista-Navarro, and Sophia Ananiadou. 2023. Argument mining as a multi-hop generative machine reading comprehension task. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10846–10858, Singapore. Association for Computational Linguistics.

Qing Lyu, Marianna Apidianaki, and Chris Callison-Burch. 2024. Towards faithful model explanation in NLP: A survey. *Computational Linguistics*, 50(2):657–723.

Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. 2023. Faithful chain-of-thought reasoning. In *The 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (IJCNLP-AACL 2023)*.

Tobias Mayer, Elena Cabrio, and Serena Villata. 2020. Transformer-based argument mining for healthcare applications. In *ECAI 2020 - 24th European Conference on Artificial Intelligence*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, pages 2108–2115. IOS Press.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online. Association for Computational Linguistics.

Hadi Mohammadi, Ayoub Bagheri, Anastasia Giachanou, and Daniel L Oberski. 2025. Explainability in practice: A survey of explainable nlp across various domains. *arXiv preprint arXiv:2502.00837*.

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.

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.

Yasser Otiefy and Alaa Alhamzeh. 2024. Exploring large language models in financial argument relation identification. In *Proceedings of the Joint Workshop of the 7th Financial Technology and Natural Language Processing, the 5th Knowledge Discovery from Unstructured Data in Financial Services, and the 4th Workshop on Economics and Natural Language Processing*, pages 119–129, Torino, Italia. Association for Computational Linguistics.

Raquel Mochales Palau and Marie-Francine Moens. 2009. Argumentation mining: the detection, classification and structure of arguments in text. In *Proceedings of the 12th International Conference on Artificial Intelligence and Law*, ICAIL '09, page 98–107, New York, NY, USA. Association for Computing Machinery.

Joonsuk Park and Claire Cardie. 2018. A corpus of erulemaking user comments for measuring evaluability of arguments. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

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

Christos Sardianos, Ioannis Manousos Katakis, Georgios Petasis, and Vangelis Karkaletsis. 2015. Argument extraction from news. In *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 56–66, Denver, CO. Association for Computational Linguistics.

Henri Savigny and Bruno Yun. 2025. Amelia: A family of multi-task end-to-end language models for argumentation. *Argument & Computation*, page 19462174251413169.

Pranjal Srivastava, Pranav Bhatnagar, and Anurag Goel. 2022. Argument mining using bert and self-attention based embeddings. In *2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, pages 1536–1540. IEEE.

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, Doha, Qatar. Association for Computational Linguistics.

Sree Harsha Tanneru, Dan Ley, Chirag Agarwal, and Himabindu Lakkaraju. 2024. On the hardness of faithful chain-of-thought reasoning in large language models. *arXiv preprint arXiv:2406.10625*.

Stephen Toulmin. 1958. *The Uses of Argument*. Cambridge University Press.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Yuhao Zhang, Jiaxin An, Ben Wang, Yan Zhang, and Jiqun Liu. 2025. Human-centered explainability in interactive information systems: A survey. *arXiv preprint arXiv:2507.02300*.

A Prompt and Human Evaluation Instruction

Prompt for our proposed method:

```
You are an expert in Argument Mining. You are
given a biomedical abstract containing numbered
argument components enclosed in <AC> tags (e.g.,
<AC1>...</AC1>).

Your task has 3 parts:

1. Classify each argument component as either a
"Claim", a "Premise".
2. Identify argument relations between argument
components in the abstract text and classify
their relation type as either "support" or
"attack".
3. For every Claim <ACi>, construct a Toulmin
structure using all Premises that relate to it:
- data: list of supporting premises
- warrant: reasoning connecting data to claim
- rebuttal: any attacking ACs

Output Format
Return a single json object with three keys:
{
  "component_types": {
    "<AC1>": "Claim" | "Premise",
    "<AC2>": "Claim" | "Premise",
    ...},
  "argument_relation_types_list": [["source AC",
  "target AC", "support" | "attack"],...,
  ["source AC", "target AC", "support" |
  "attack"]],
  "toulmin_structures": {
    "<ACi>": {
        "data": ["...", "..."],
        "warrant": "...",
        "rebuttal": ""
    }
}
"argument_relation_types_list": the relation
types between different argument components,
"source AC" and "target AC" are argument
components (ACs) in the abstract text (such as
"<AC1>", "<AC2>", ...).

Do not include any explanations or extra text,
just the json object.
```

Prompt for the component wise baseline for ACC task:

```
Task: Argument Component Classification (ACC).
Given a single argument component extracted from
a paragraph, predict its type. Allowed labels:
{claim, premise}. Return only the label.
{input_text}
```

Prompt for the pairwise baseline for ARIC task:

```
Task: Argument Relation Identification &
Classification (ARIC). Given a pair of argument
components (source, target) from the same
paragraph, predict the relation type between
them. Allowed labels: {support, attack, none}.
Return only the label.
source: {source_text}, target: {target_text}.
```

Prompt for the end-to-end baseline:

```
You are an expert in Argument Mining. You are
given a biomedical abstract containing numbered
argument components enclosed in <AC> tags (e.g.,
<AC1>...</AC1>).
Your task is to classify each argument component
as either a "Claim", a "Premise" and identify
argument relations between argument components
in the abstract text and classify their relation
type as either "support" or "attack".
Output Format
```

```
Return a single json object with two keys:
{
  "component_types": {
    "<AC1>": "Claim" | "Premise",
    "<AC2>": "Claim" | "Premise",
    ...}
  "argument_relation_types_list": [["source AC",
  "target AC", "support" | "attack"],...,
  ["source AC", "target AC", "support" |
  "attack"]]
}
"argument_relation_types_list": the relation
types between different argument components,
"source AC" and "target AC" is argument
components (ACs) in the abstract text (such as
"<AC1>", "<AC2>", ...).

Do not include any explanations or extra text,
just the json object.
```

Prompt for the CoT baseline:

```
You are an expert in Argument Mining. You are
given a biomedical abstract containing numbered
argument components enclosed in <AC> tags (e.g.,
<AC1>...</AC1>).
Your task is to classify each argument component
as either a "Claim", a "Premise" and identify
argument relations between argument components
in the abstract text and classify their relation
type as either "support" or "attack".
Let's think step by step.

Output Format
Return a single json object:
{
  "reasoning_path": Your reasoning process,
  "component_types": {
    "<AC1>": "Claim" | "Premise",
    "<AC2>": "Claim" | "Premise",
    ...}
  "argument_relation_types_list": [["source AC",
  "target AC", "support" | "attack"],...,
  ["source AC", "target AC", "support" |
  "attack"]]
}
"reasoning_path": Your reasoning process.
"argument_relation_types_list": the relation
types between different argument components,
"source AC" and "target AC" is argument
components (ACs) in the abstract text (such as
"<AC1>", "<AC2>", ...).

Do not include any explanations or extra text,
just the json object.
```

Prompt for usefulness evaluation

```
You are an expert evaluator of explanation
usefulness for argument-mining predictions.

Evaluate explanations based on their ability to
accurately justify the predicted argument
components and relations, not based on
plain-language fluency or narrative style.

Your goal is to evaluate Explanation 1 and
Explanation 2 strictly based on the following
usefulness criteria.

### Usefulness Criteria (score each 1–5):
1. **Practical usefulness** – how well the
explanation helps a user understand the predicted
components and relations.
2. **Actionability** – how well the explanation
helps a user apply, verify, or critique the
prediction.
4. **Trust support** – how well the explanation
helps a user judge the reliability of the
prediction without adding hallucinated content.

### INPUT FIELDS:
1. input text: {input_text}

2. **Argument Mining Prediction 1**
{argument_mining_prediction}

3. **Explanation 1**
{explanation_1}

4. **Argument Mining Prediction 2**
{argument_mining_prediction_2}

5. **Explanation 2**
{explanation_2}

### Output Format (STRICT):
{{
  "scores": {{
    "explanation_1": {{
      "practical_usefulness": X,
      "actionability": X,
      "trust_support": X
    }},
    "explanation_2": {{
      "practical_usefulness": X,
      "actionability": X,
      "trust_support": X
    }}
  }},
  "winner": "explanation_1" | "explanation_2" |
  "Tie"
}}

The "winner" must be chosen by comparing total
scores.

Do not add any explanations.
```

Human evaluation instruction:

```
Human evaluation instructions
1. Overview of the Task

In this task, you will evaluate how useful two
explanations are for two AI predictions about the
same biomedical abstract.

Each example includes:

    Input text – a biomedical abstract labeled
    with argument components (<AC1>, <AC2>,
    . . . )
    Prediction - component types + argument
    relations
    Explanation – a Toulmin-structured
    explanation of this prediction

    Prediction  – another model's component types
    + argument relations
```

Explanation — a free-form (narrative) explanation of this prediction

Your task is to evaluate the usefulness of each explanation relative to its own prediction.

You will score both explanations on three criteria:

- Practical Usefulness
- Actionability
- Trust Support

Then you will select which explanation is overall better (or mark them as a tie).

2. Understanding the Input Text (Abstracts)

Each abstract is annotated with AC tags:
<AC1> ... </AC1>
<AC2> ... </AC2> ...

Each AC describes one argument component in the abstract.
Predictions assign each AC a type:

- Claim
- Premise

And relations:

- support
- attack

You will not evaluate the correctness of the prediction itself.
You only judge how well each explanation helps you understand that prediction.

3. How to Read a Toulmin Explanation

One of the explanations is usually written in a Toulmin structure.
For each claim, the model presents a block of the form:

Example format:

CLAIM: <AC1>
• Data: <AC2>, <AC5>
• Warrant: Statistically significant improvements support the beneficial effect.
• Rebuttal: (none)
What this means:

- Claim (top-level AC tag)
  The AC tag shown at the top (for example, <AC1>) is the claim being explained.
- Data
  These are AC components the model believes provide evidence supporting that claim.
- Warrant
  This is the reasoning that explains why the data supports the claim.
- Rebuttal
  These are AC components that weaken or contradict the claim.
  This may be empty if the model found no rebuttal evidence.

Multiple Toulmin Blocks

If several top-level ACs are listed one after another, such as:

CLAIM: <AC1>
. . .

CLAIM: <AC7>
. . .

CLAIM: <AC8>
. . .

This means the model has produced one Toulmin explanation per claim in Prediction 1.

Each block is separate and explains a different claim in the model's prediction.

4. How to Read a Narrative explanation

One of the explanations is a paragraph or structured text that:

- describes what each AC means
- describes how ACs relate
- summarizes why the prediction is sensible

It is more free-form and less structured than a Toulmin block.

Your job is to evaluate how useful it is—not how well it is written.

5. Important: Each Explanation Belongs to Its Own Prediction

- Explanation 1 should be evaluated only with respect to Prediction 1.
- Explanation 2 should be evaluated only with respect to Prediction 2.

Do not compare the two predictions.
You compare how useful the explanations are.

Each explanation is attempting to justify its own prediction

6. Evaluation Criteria (1–5 scale)

For each explanation, you will assign three scores.

Score the content, not the format.
Both free-text explanations and structured explanations (e.g., Toulmin: data–warrant–rebuttal) can achieve any score.

Criterion 1 — Practical Usefulness

Question:
Does this explanation help you understand the model's predicted components and relations?

You should look for whether the explanation — in whatever style it uses — makes the predicted structure easier to understand.
How to score

5 — Excellent

- Very clear and easy to follow
- Explicitly connects reasoning to the input content
  - This may be done through AC tags, paraphrased evidence, or structured fields
- No unnecessary information

4 — Good

- Mostly clear and helpful
- Shows correct understanding, but may miss a minor connection or detail

3 — Adequate

- Somewhat helpful but incomplete
- Includes partial or shallow explanations
- Mentions relevant content but lacks full clarity

2 — Poor

- Hard to follow or too vague
- Missing most explanations of component types or relations

1 — Very Poor

- Does not help you understand the prediction at all
- Irrelevant, incorrect, or purely a summary

Criterion 2 — Actionability

Question:
Does the explanation help you verify, check, or critique the prediction?

An explanation is actionable if it gives you enough information to confirm or question the prediction — regardless of whether this is done in structured form or free text.
How to score

5 — Excellent

- Makes verification straightforward
- Points to specific evidence or reasoning
- Clearly shows how the prediction could be checked

4 — Good

- Mostly checkable
- Provides useful evidence or reasoning but not fully complete

3 — Adequate

- Partially checkable
- Provides some reasoning but insufficient to assess the full prediction

2 — Poor

- Hard to verify
- Evidence or reasoning is minimal or unclear

1 — Very Poor

- Not verifiable at all
- No usable reasoning or evidence

Criterion 3 — Trust Support

Question:
Does the explanation help you judge whether the prediction is reliable and grounded?

A trustworthy explanation is grounded, faithful, and avoids unsupported claims.
This applies equally to free-text and structured explanations.
How to score

5 — Excellent

- Fully grounded in the provided content
- No hallucinations or invented facts
- Describes relations correctly

4 — Good

- Mostly grounded and correct
- Minor omissions or imprecise wording

3 — Adequate

- Generally grounded but incomplete
- Some shallow reasoning or missing detail

2 — Poor

- Contains inaccuracies or unsupported statements
- Vague or weak grounding

1 — Very Poor

- Hallucinates or significantly misrepresents the content
- Cannot be trusted

7. Overall Winner (Choose One)

After rating both explanations:

- Add the three scores for Explanation 1
- Add the three scores for Explanation 2

Then choose:

- Explanation 1 is better, or
- Explanation 2 is better, or
- Tie

```
You are evaluating usefulness, not grammatical
quality or prediction correctness.


8. What You MUST NOT Do

     Do not compare Prediction 1 vs Prediction 2
     directly
     Do not evaluate scientific truth of the
     abstract
     Do not reward writing style or fluency
     Do not judge whether the model's labels are
     correct

Your job is to evaluate explanation usefulness
only.


9. Quick Annotator Checklist

Before submitting each example:

     I read the abstract
     I read Prediction 1 + Explanation 1
     I read Prediction 2 + Explanation 2
     I rated each explanation on all 3 criteria
     I compared explanations only on usefulness
     I selected the correct overall winner

10. Time Expectation

    Each example: ~2-3 minutes
    Total 50 examples: ~100-150 minutes
```

B Dataset licenses and statistics

We use two public available datasets for our experiments: AbstRCT (Mayer et al., 2020) and CDCP (Park and Cardie, 2018). The datasets have the following licenses.

AbstRCT: Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0).
CDCP: ODC Open Database License v1.0 [12,13].

All datasets are in English.

Table 6 and Table 7 shows the train test split for these datasets.

split	num
train	350
gla test	100
mix test	100
neo test	100

Table 6: Train test split for abstRCT datset.

split	num
train	580
test	150

Table 7: Train test split for CDCP dataset.

C Prompting configurations

We use greedy decoding for all the prompting experiments (`temperature = 0.0, top_p = 1.0`). Gemma models are accessed using Google AI studio, all the other models are accessed through the OpenRouter platform (https://openrouter.ai), which provides unified access to multiple model providers. The model, provider and identifier are shown in Table 8.

D Usefulness evaluation: additional results

Agreement between different judges:
Qwen vs. DeepSeek: raw = 0.84, $\kappa = 0.286$,
Qwen vs. GPT: raw = 0.84, $\kappa = 0.286$,
DeepSeek vs. GPT: raw = 0.96, $\kappa = 0.479$.

E Human evaluation details

Two colleagues voluntarily contributed annotations for the usefulness evaluation after being informed that their ratings would be used for research publication; no compensation was provided, and no personal data were collected. The evaluation instruction is provided in Appendix A.

F Additional qualitative examples

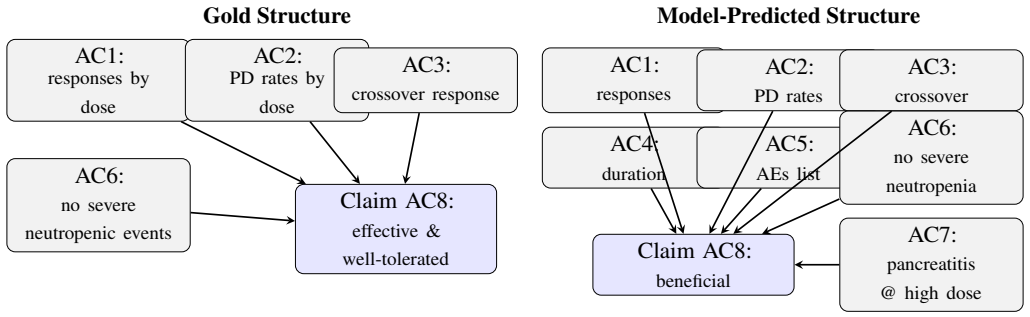
[12] http://www.opendatacommons.org/licenses/odbl/1.0/
[13] https://spdx.org/licenses/ODbL-1.0.html

Model	Provider	Identifier
Gemma 3 27B	Google AI studio	`gemma-3-27b-it`
LLaMA 3.3 70B	`deepinfra/turbo`	`meta-llama/llama-3.3-70b-instruct`
Qwen 3 235B	`deepinfra/fp8`	`qwen/qwen3-235b-a22b-2507`
DeepSeek V3.2	`ionstream/fp8`	`deepseek/deepseek-v3.2`
GPT 4.1	`openai`	`openai/gpt-4.1`

Table 8: Provider and identifier for the models we use in the prompting experiments.

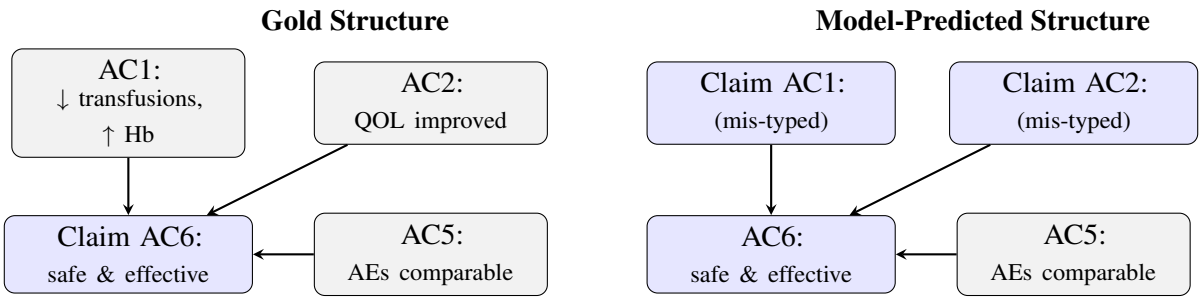
Evaluator	Wins (Toulmin/CoT)	Avg Total Score (Toulmin)	Avg Total Score (CoT)	Mean $\Delta(T - C)$	95% bootstrap CI
Annotator 1	37 / 12	12.14	9.10	+3.04	[1.62, 4.46]
Annotator 2	37 / 10	12.58	10.54	+2.06	[1.07, 3.05]
Qwen 3 235B	40 / 10	13.84	10.58	+3.26	[2.12, 4.32]
DeepSeek V3.2	48 / 2	13.36	5.40	+7.96	[6.96, 8.80]
GPT 4.1	48 / 2	14.52	9.06	+5.46	[4.74, 6.06]

Table 9: Usefulness comparison between Toulmin and CoT explanations.



Toulmin explanations for AC8
Data: AC1–AC7 (responses, PD, duration, AEs, no severe neutropenia, pancreatitis @ high dose).
Warrant: *"Responses and tolerability indicate benefit."*
Rebuttal: (none).

Figure 6: (bexarotene/CTCL): Gold vs Predicted. Gold supports AC8 with a subset; the model predicts nearly all ACs as supporters for AC8. The Toulmin structures accurately reflect this and provide insights into why the model makes these errors.



Toulmin Explanation

Claim AC2 (model mis-typed as Claim)
Data: *[empty]*
Warrant: *"Significant improvement in QOL domains indicates a positive effect."*

Claim AC6 (safe & effective)
Data: AC1 (↓ transfusions, ↑Hb), AC2 (QOL↑), AC5 (AEs comparable).
Warrant: *"Reductions in transfusions, hemoglobin gains, and QOL improvements support safety and efficacy."*

Error Revealed: The model mis-classifies AC2 as a **Claim** but provides **no data** for it, indicating there may be errors in model predictions. This asymmetry (AC6 well-supported vs. AC2 unsupported) is only visible through the Toulmin structure, not ACC/ARIC.

Figure 7: (epoetin alfa): Gold vs Model and its Toulmin explanation. The model predicts AC2 as a claim but provides no data in the toulmin explanations.