Assessing the Minimal Dialectical Quality in Argumentation: A Neuro-Symbolic Approach Integrating Argument Mining, Quality Assessment, and Probabilistic Reasoning

VICTOR HUGO NASCIMENTO ROCHA*, Universidade de São Paulo, Brazil
FABIO GAGLIARDI COZMAN†, Universidade de São Paulo, Brazil
SERENA VILLATA, Université Côte d'Azur, CNRS, Inria, I3S, France

This work introduces a new dimension of argumentative quality, termed Minimal Dialectical Quality (MDQ), which requires an argumentation to coherently support its main claims through adequate justifications and, when necessary, explicit rebuttals. MDQ provides a less subjective notion of argumentation quality than many existing approaches, as it relies exclusively on information contained in the text itself and does not require external knowledge or fact-checking. To investigate the feasibility of MDQ as an approximation of human judgments of argumentative quality, we propose the Minimal Dialectical Quality Evaluator (MiDiQE), a neuro-symbolic system for assessing the quality of argumentative essays. MiDiQE operationalizes MDQ by computing the probability that a given text satisfies the new dimension and comparing it against a predefined threshold. The system integrates neural Argumentation Mining, sub-symbolic Single Argument Quality assessment, and symbolic Argumentation Reasoning modules to extract argument structures, evaluate their components, and perform formal probabilistic reasoning over them. Experimental results on argumentative essay datasets show that MDQ aligns well with human quality assessments, supporting its validity as a meaningful quality dimension. They also demonstrate that MiDiQE produces interpretable evaluations of argumentative quality while highlighting both the strengths and limitations of the proposed approach. Potential applications and directions for future work are discussed.

JAIR Associate Editor: Roberta Calegari

JAIR Reference Format:
Victor Hugo Nascimento Rocha, Fabio Gagliardi Cozman, and Serena Villata. 2026. Assessing the Minimal Dialectical Quality in Argumentation: A Neuro-Symbolic Approach Integrating Argument Mining, Quality Assessment, and Probabilistic Reasoning. *Journal of Artificial Intelligence Research* 86, Article 45 (August 2026), 55 pages. DOI: doi: 10.1613/jair.1.21868)

1 Introduction

Argumentation is a fundamental aspect of human interaction, playing a key role in justifying claims, challenging ideas, persuading colleagues. It is central to disciplines such as law and politics and is generally understood as a process of divergence, where parties seek to persuade each other through reasoned exchanges until consensus is reached or dialogue ceases (Eemeren 2018; Toulmin 2003). In artificial intelligence (AI), computational argumentation has emerged as a key research area when creating systems that can explain their reasoning in a way that is understandable to humans (Cocarascu et al. 2019).

*Corresponding Author.
†Corresponding Author.

Authors' Contact Information: Victor Hugo Nascimento Rocha, ORCID: 0000-0002-8984-7982, victor.hugo.rocha@usp.br, Universidade de São Paulo, São Paulo, SP, Brazil; Fabio Gagliardi Cozman, ORCID: 0000-0003-4077-4935, fgcozman@usp.br, Universidade de São Paulo, São Paulo, SP, Brazil; Serena Villata, ORCID: 0000-0003-3495-493X, serena.villata@cnrs.fr, Université Côte d'Azur, CNRS, Inria, I3S, France.

DOI: doi: 10.1613/jair.1.21868)

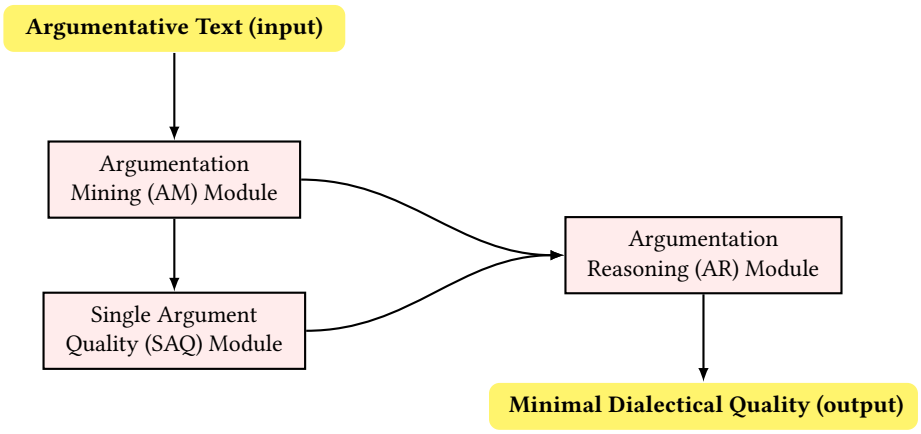

Fig. 1. Overview of the MiDiQE system.

A central issue in argumentation is assessing the quality of arguments, an ability that is particularly important in fighting misinformation and improving the reliability of AI-generated text (Rocha, Silveira, et al. 2023). However, while argument quality assessment has seen significant progress, a robust method for evaluating the overall quality of argumentation within a text remains a challenge.

To address this gap, this paper introduces the Minimal Dialectical Quality (MDQ), a novel dimension for assessing argumentation quality that requires an argumentative structure to coherently support its main claims through adequate justifications and, when necessary, explicit rebuttals. MDQ conveys a dialectical measure of argumentation quality that is independent of rhetorical effectiveness or factual correctness, relying solely on information contained in the text itself.

To operationalize and empirically evaluate this new dimension, we propose the Minimal Dialectical Quality Evaluator (MiDiQE), a neuro-symbolic system designed to assess argumentative essays by estimating the probability that their argumentation satisfies MDQ and comparing it to a predefined threshold. MiDiQE serves both as an application of MDQ and as a means to investigate whether it is able to capture human perceptions of argumentative quality in texts.

As shown in Figure 1, MiDiQE integrates three key areas of computational argumentation: (1) Argumentation Mining (AM), which extracts argumentative structures from text, (2) Single Argument Quality Assessment (SAQ), which evaluates the strength of individual arguments, and (3) Argumentation Reasoning (AR), which applies formal reasoning to determine whether the argumentation structure successfully supports its conclusions, by producing a probability of minimal dialectical qualify for an input argument. Together, these components enable MiDiQE to produce structured, interpretable assessments of argumentation quality. Experiments on argumentative essay datasets demonstrate the system's effectiveness and the MDQ's suitability as a (necessarily approximate) measure of human judgments about argumentative quality.

1.1 Contributions

This work makes the following contributions:

(1) Minimal Dialectical Quality: A novel dimension for evaluating argumentation quality based on a text's ability to defend its main points and that is shown to produce good approximations about human perceptions on that quality;

(2) Minimal Dialectical Quality Evaluator: A new neuro-symbolic system that integrates AM, SAQ and AR and is designed to evaluate, through a combination of sub-symbolic and symbolic (logical and probabilistic) modules, the argumentative quality of essays;
(3) Experimental Validation: A study that demonstrates the MiDiQE's effectiveness in measuring MDQ and the dimension's ability to produce good approximations about human judgments on argumentative quality;
(4) Secondary Contributions:
   - In AM: A multi-domain dataset unifying annotation schemes, a graph neural network model, and a new simple way to use the linguistic context of arguments;
   - In SAQ: A novel preprocessing strategy for argument data that implicitly encodes the structure of the argumentation graph, thus enabling a Random Forest model to access metrics such as Cogency and Reasonableness;
   - In AR: An implementation of the L-Credal semantics (Rocha and Cozman 2022a) within the the Bipolar Conclusion-Augmented Argumentation Framework (BCAF) (Rocha and Cozman 2022b) through the DPASP package (Geh et al. 2023).

1.2 Paper Structure

This paper is structured as follows: Section 2 introduces the problem of assessing argumentation quality, proposes the MDQ dimension, the MDQ criterion (MDQC) and the MiDiQE system. It also details the datasets and methodology used for experimentation. Section 3 presents the sub-symbolic modules of MiDiQE. It begins with the neural AM module, discussing its design, implementation, and experimental results, with a focus on graph neural networks and a multi-domain dataset. It then introduces the SAQ module, outlining its metrics and integration into the system. Section 4 describes MiDiQE's symbolic module, i.e. its AR module. It includes an implementation of the BCAF and the L-credal semantics for probabilistic reasoning. Section 5 consolidates the modules into a unified system, evaluates its performance in assessing the Minimal Dialectical Quality of essays, and validates the quality dimension against human judgments. Finally, Section 6 concludes the paper, summarizing contributions and discussing future research directions.

2 A Minimal Criterion for the Dialectical Evaluation of Argumentation Quality

In this section, we review the literature on argumentation quality assessment in artificial intelligence, introduce a new criterion for argumentation quality, and describe the methodology and datasets used to develop a system capable of assessing this criterion.

Before proceeding, however, it is important to clarify how the terms *(single) argument quality* and *argumentation quality* are used in our paper, as we deliberately adopt a distinction between them.

In the context of argumentative essays—our chosen application domain—argument quality refers to the assessment of the strength or adequacy of an individual argument or of specific argumentative relations within a given context. As such, it is a local notion: it evaluates components of an argumentative text in isolation, without considering the overall structure of the argumentation as a whole.

By contrast, argumentation quality concerns the assessment of the entire argumentative discourse, taking into account all arguments and the relations among them. This notion is inherently global. We do not claim that this distinction is universally adopted or applicable across all domains. However, we find it both meaningful and practically useful for argumentative essays, where local assessments of individual arguments and global assessments of the overall argumentative structure serve complementary but distinct purposes.

2.1 Quality Assessment: Background

The study of argumentation quality within AI has evolved through several research efforts. The proposals in this field are so varied, that O'Keefe and Jackson (1995) asserted that neither a general idea of what argument(ation) quality in natural language is nor a clear definition of its dimensions can exist. However, a notable attempt to create a unified conceptualization to cover many approaches in the quality assessment of argumentation is the work by Wachsmuth, Naderi, et al. (2017). The authors propose a comprehensive framework that consolidates the diverse concepts of argumentation quality into three primary dimensions, each representing fundamental qualities essential for evaluating arguments: logical cogency, dialectical reasonableness, and rhetorical effectiveness. These dimensions are further broken down into sub-dimensions, producing a detailed 15-dimensional quality model in which *Argumentation Quality* serves as the overarching category. The dimensions that compose it are defined as follows:

- **Cogency**: evaluates an argument's logical soundness. Cogency operates more on the level of a single argument's quality rather than on the whole argumentation. For an argument to be cogent, it must have:
  (1) Local acceptability: The premises should be credible and rationally believable;
  (2) Local relevance: Each premise should contribute directly to accepting or rejecting the argument's conclusion;
  (3) Local sufficiency: The premises, when taken together, should provide enough support to make the conclusion rationally defensible.
- **Reasonableness**: assesses the argument's contribution to resolving the issue at hand. Reasonableness operates both in the single argument level as well as in the whole of argumentation. An argument(ation) is deemed reasonable with respect to:
  (1) Global acceptability: The audience finds the argument(ation) acceptable as a fair and valid contribution to the issue;
  (2) Global relevance: The argument(ation) offers information that moves toward a resolution, helping guide the audience to an ultimate conclusion;
  (3) Global sufficiency: The argument(ation) anticipates and sufficiently rebuts potential counterarguments to support its stance.
- **Effectiveness**: examines the rhetorical appeal and clarity of the argument, including its ability to persuade and engage the audience, often influenced by emotional or stylistic elements. Effectiveness includes:
  (1) Credibility: The argument should present information in a way that builds trust in the author's reliability;
  (2) Emotional appeal: Persuasive arguments effectively evoke emotions, making the audience more receptive to the author's claims;
  (3) Clarity: Clear and straightforward language is crucial, avoiding unnecessary complexity or ambiguity;
  (4) Appropriateness: The style and tone of language should enhance credibility and emotional impact, while remaining suitable to the issue;
  (5) Arrangement: Properly arranged arguments introduce the issue, present supporting points and, in that order, their conclusion.

To avoid confusion, "dimension" in this context refers not to a mathematical or formal notion of dimensionality but to a focus of analysis. For instance, overall argumentation quality is the goal of the Argumentation Quality dimension of Wachsmuth, Naderi, et al.'s (2017) taxonomy, while assessing the credibility of individual premises corresponds to the Local Acceptability dimension.

Given this 15-dimensions taxonomy, the authors present a dataset of 320 arguments, each annotated across their dimensions. They emphasize that this taxonomy not only establishes a common ground for assessing argument quality computationally but also bridges multiple evaluative frameworks within argumentation theory, uniting rhetorical, dialectical, and logical perspectives.

Several efforts in the literature can be classified by their focus on what was defined by Wachsmuth, Naderi, et al. (2017) as Effectiveness. Stahl et al. (2024) and Wachsmuth, Al-Khatib, et al. (2016) explore the relationship between argumentative structure and quality, demonstrating how organizational features such as argumentative discourse units (ADUs) and flow patterns influence perceived quality. Similarly, work such as by El Baff et al. (2018), Persing and Ng (2015) and Toledo et al. (2019) emphasize rhetorical effectiveness, focusing on audience impact and persuasive strength and using criteria such as prior reader beliefs, linguistic choices, psychological impact of the text on reader and averages of human annotations to evaluate quality.

On the other hand, approaches such as by Cabrio and Villata (2012) and Wachsmuth, Stein, et al. (2017) include dialectical elements, i.e. are more in line with the Reasonableness dimension. That work uses less subjective criteria to evaluate the quality of arguments. Notably, Cabrio and Villata (2012) combine Textual Entailment (TE) with abstract argumentation frameworks to analyze online discussions and to identify widely accepted stances, measuring argument quality by its level of acceptance within an argumentation framework. Similarly, Wachsmuth, Stein, et al. (2017) apply a graph-based model inspired by the PageRank algorithm, where the "importance" of an argument is measured by the frequency of its reuse as a premise in other arguments.

Interestingly, a more recent work by Marro et al. (2022) integrates all three dimensions of Wachsmuth, Naderi, et al.'s taxonomy. The authors annotate a dataset of 402 persuasive student essays, derived from Stab and Gurevych (2017), with quality attributes across three dimensions: Cogency, Rhetoric, and Reasonableness. Each essay is represented as an argument graph where nodes represent argumentative components (major claims, claims and premises) and edges represent support or attack relationships. Each major claim and claim, which can be understood as the conclusion of each argument, are annotated, taking into account their premises, for the Cogency score, the strength of the attacks they receive, for the Reasonableness scores, and other features for the Rhetoric metric. In addition, the authors employ graph embeddings (FEATHER-G) and text embeddings (Longformer) to predict quality scores for individual arguments. These previous efforts however primarily focus on individual arguments rather than the overall quality of argumentation.

Beyond the taxonomy highlighted so far, there are other proposals in the literature. The rise of Large Language Models (LLMs) has sparked interest in their potential for argument quality assessment. In a recent study, Wachsmuth, Lapesa, et al. (2024) explore the application of LLMs to measure quality in arguments. They claim that while LLMs show promise in detecting quality aspects such as coherence and clarity, there are several challenges in using them for nuanced, context-sensitive judgments and also ethical and social considerations to take into account, since often LLMs reflect negative social biases.

Finally, it is also worth noting that the field of Automatic Essay Scoring (AES) (Lagakis and Demetriadis 2021) intersects with the Argument Quality Assessment literature, given that essays with high scores are those that meet basic requirements, such as presenting a clear and concise claim, and providing strong and logical premises to support it. AES however tends to evaluate the quality of essays through indirect means, such as word count and topic distribution similarity (Zhang and Litman 2021), while AQ seeks more specific and deep argumentative metrics.

2.2 The Minimal Dialectical Quality of Argumentation

To develop a computational system for evaluating argumentation quality, this work introduces a new quality dimension that both aligns with and extends Wachsmuth, Naderi, et al.'s (2017) taxonomy: the Minimal Dialectical Quality of Argumentation. Positioned on the dialectical side of the taxonomy, the MDQ dimension builds on Cogency factors while also taking into account an argument's ability to rebut attacks, excluding rhetorical aspects.

This new dimension adopts a single requirement for an argumentation to have high quality: it must be internally coherent. That is, without relying on external fact-checking or audience considerations, the argumentation must

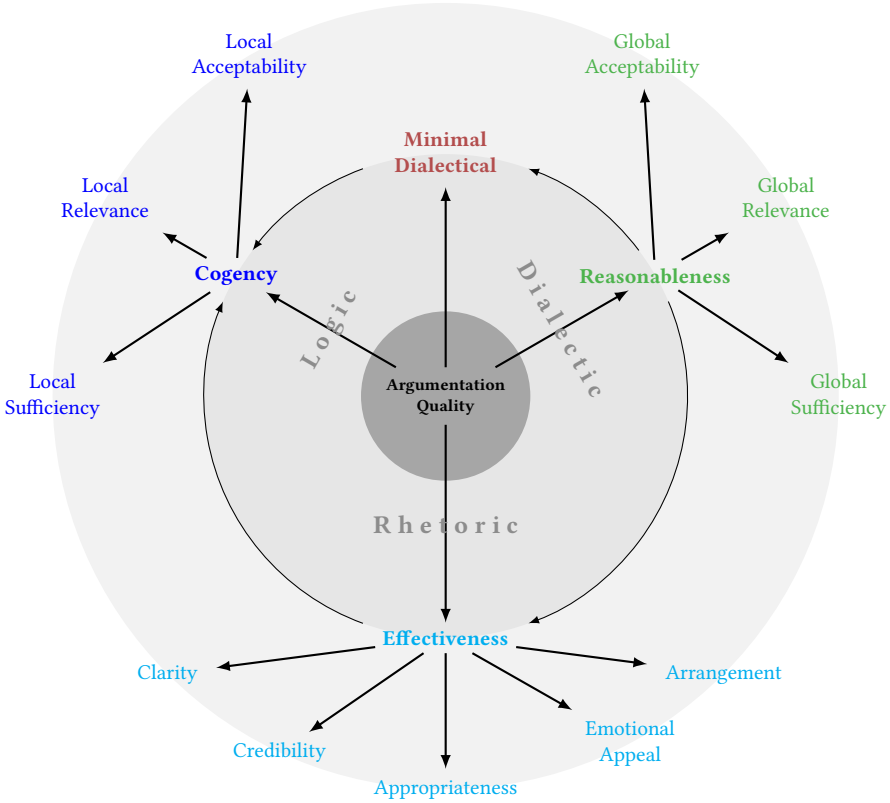

Fig. 2. Expanded taxonomy for Argumentation Quality including the concept of Minimal Dialectical Quality. This new dimension depends on Cogency, takes into account Reasonableness of individual arguments and does not consider Rhetorical features.

be able to sustain its main points based solely on the information it presents. Only when this condition is met can the higher-level Dialectical Reasonableness sub-dimensions be meaningfully assessed.

This dialectical minimalist dimension is defined as:

Definition 1. *The* Minimal Dialectical Quality *of an argumentation assesses whether a text can internally support its main points by presenting cogent arguments and by effectively rebutting counterarguments when these are present.*

The rationale for the Minimal Dialectical Quality comes from the fundamental definition of argumentation as a process of defending or rejecting a proposition p by providing reasons to support it and countering objections. The MDQ evaluates whether an argumentation structure can internally justify its main points without relying on external validation or rhetorical appeal. The dimension addresses a gap in the literature by providing a systematic, less subjective basis for evaluating the overall quality of an argumentation. Unlike existing approaches, which often focus on individual arguments or rhetorical effectiveness, the Minimal Dialectical Quality dimension emphasizes the structural and logical integrity of the entire argumentation. On the one hand, even one who is prone to judge an argumentation as deficient because of one's disagreement with the points made, would have difficulty objecting to its minimal quality based only on the arguments explicitly made. On the other hand,

even the strongest supporter of a given point of view would have trouble defending an argumentation that lacks internal structure and/or is self-contradictory.

Figure 2 illustrates the 15-dimensional taxonomy of argumentation quality as presented in the original proposal, now expanded to include the new Minimal Dialectical Quality dimension. The circular arrows indicate that MDQ, just as Effectiveness, depends on Cogency, while Reasonableness depends on both the new dimension and Effectiveness. The additional arrows further divide Argumentation Quality into its main dimensions and their respective sub-dimensions.

As the Figure suggests, MDQ can only be fully evaluated when the Cogency of individual arguments is considered. At the same time, MDQ goes beyond Cogency, because it also accounts for counterarguments and rebuttals explicitly present in the text, rather than only the premises supporting a claim. However, MDQ also differs from Reasonableness in that it focuses exclusively on information contained within the text itself, without requiring external knowledge to determine whether the argumentation contributes to resolving the issue at hand. For this reason, MDQ is positioned between Cogency and Reasonableness within the taxonomy: it extends beyond purely logical evaluation by incorporating explicit dialectical interactions, while still remaining restricted to the minimal dialectical information contained within the text itself—hence the name *Minimal Dialectical Quality*.

Figure 2 also highlights that the new dimension is not expected to be sufficient to assess all the dimensions of argumentative quality. For that, it would have to be combined with other criteria such as the others detailed before. The proposal here is that this new quality dimension is the dialectical *minimal* standard an argumentation must achieve; thus, it complements the full analysis of quality proposed in other works by providing a less subjective basis for evaluation and allowing quality to be measured using only the text itself, without the need for external knowledge or fact-checking.

This distinctive feature of MDQ—namely, its exclusive reliance on information contained in the text itself—is precisely what enables its computational realization in this work. Because MDQ does not depend on external knowledge or fact-checking, it supports automated assessments that serve as meaningful approximations of human judgments of argumentative quality. This practical applicability constitutes one of MDQ's main advantages over other quality dimensions proposed in the literature.

Before describing how MDQ is applied computationally, we first introduce a more formal criterion that instantiates this new dimension of argumentative quality.

Definition 2. *The* Minimal Dialectical Quality Criterion *assesses whether the argumentation in a given text can be considered good by comparing the probability that MDQ holds for that text against a predefined threshold.*

Our system that validates the MDQC, the Minimal Dialectical Quality Evaluator, is, in fact, designed to assess the probability $\mathbb{P}(\text{MDQ})$ attached to an input text and then apply the criterion based on a predefined threshold. As shown in Figure 1,MiDiQE first extracts the argumentative structure of a natural language text using neural techniques. It then evaluates individual arguments for Cogency and Reasonableness through a sub-symbolic process before passing them to a symbolic module, which determines whether the argumentative structure sufficiently supports the text's main claims (and the probability of it). By combining sub-symbolic and symbolic reasoning, MiDiQE assesses (the probability of) the minimal dimension using only the information contained within the text.

To verify the practical validity of the approximations generated by MDQC and the MiDiQE system, we must test whether MDQ correlates well with the quality of argumentation as it is understood/annotated by human beings. We focus on the argumentative essay domain to test our proposals. There are two reasons for this. Firstly, this type of text is inherently argumentative with a very well-defined structure. Secondly, argumentative essays, in particular essays written by students, are one of the few types of text studied in the literature for computational argumentation that have annotations of argumentation quality, of argumentation structure and of strength of individual arguments.

In short, the implemented neuro-symbolic MIDIQE system was tested in two ways to validate both itself and the MDQC. First, we evaluated whether the system can in fact capture the minimal quality dimension and its criterion. Second, the performance of the system in a dataset annotated for quality using different criteria was assessed to quantify the correlation between the minimal criterion and quality as a whole. The next section introduces the datasets used in each of those experiments.

2.3 Methodology and Datasets for Argumentation Quality Evaluation

The development and validation of MIDIQE require suitable data that support neural/sub-symbolic model training and provide ground truth values for argumentation quality. Two types of quality annotations are essential: (1) datasets with explicit MDQ annotations to validate MIDIQE's accuracy in measuring it, and (2) datasets with human-annotated quality scores, that can be used to assess how the MDQ correlates with human judgments.

We describe two datasets that we employed in this work in Sections 2.3.1 and 2.3.2. The AAEC dataset (Stab and Gurevych 2017) is the only existing resource that fully meets the requirements for explicit MDQ annotations. It provides training data for MIDIQE's AM and SAQ sub-symbolic models and can be preprocessed to incorporate MDQ-based quality annotations. Therefore, it was selected for this study, with its details and preprocessing steps outlined in Section 2.3.1. The ArGPT dataset (Rocha, Silveira, et al. 2023) is, to the best of our knowledge, the only available resource for human-perceived quality assessments within argumentative essays and AM annotations. Its key characteristics are described in Section 2.3.2.

Finally, to ensure a comprehensive evaluation of MIDIQE scores for quality, both regression and classification metrics were selected. For the first, the numerical quality scores available in both datasets were used, while for the latter, we resorted to a binary system with *Good* and *Bad* labels, since real-world applications – such as misinformation detection – may require an interpretable approach. The evaluation methods for both tasks are detailed in Section 2.3.3.

2.3.1 AAEC Dataset. The AAEC dataset was developed by Stab and Gurevych (2017) to address the growing need for annotated resources for argument mining tasks. It consists of 402 persuasive essays written by high school students on a variety of topics. Each essay was broken down into argumentative discourse units (ADUs), representing the components of the argument. These components were annotated as: (1) *Major Claims*: representing the central stance, or main point, of the essay; (2) *Claims*: which support or challenge the major claim; (3) *Premises*: which provide evidence or reasons backing the claims or major claims.

The relations between these components were also annotated, distinguishing between support and attack relations. For instance, a premise might support a claim by providing evidence, while a claim might challenge the major claim of the essay. This results in a tree-like structure, with the major claim as the root and other components as the branches.

The annotation process adhered to rigorous guidelines to ensure high inter-annotator reliability. Disagreements were resolved through iterative discussions, leading to a gold standard dataset for argumentation mining tasks.

Building upon this, Marro et al. (2022) expanded the dataset to include its three dimensions of argument quality, based on Wachsmuth, Naderi, et al. (2017) dimensions: Cogency, Reasonableness - which is further divided into Reasonableness Counterargument and Reasonableness Rebuttal - and Rhetoric. The annotation of these quality dimensions followed a set of scoring rubrics developed by the authors, which incorporated principles from social sciences and argumentation theory, particularly the work of Stapleton and Wu (2015). Annotators assigned scores on a scale of 0, 10, 15, 20, and 25 for each dimension.

Although the AAEC dataset is not directly annotated for Minimal Dialectical Quality (MDQ), its existing annotations let us extract sensible values of $\mathbb{P}(\text{MDQ})$, the probability of minimal dialectical quality for a given input argument. MDQ focuses on an essay's ability to support its main points, thus MDQ is derived from the

Major Claim's annotations for Cogency (C), Reasonableness Counterargument (RC), and Reasonableness Rebuttal (RR), excluding the Rhetoric metric. Therefore, the calculation of the $\mathbb{P}$(MDQ) metric itself was done as follows:

(1) We took the original scale of the annotations (0, 10, 15, 20, and 25) in AAEC, and multiplied its values by 4 to convert them to a 0 to 100 range;
(2) We observed that the Cogency value of a Major Claim was originally annotated considering only the supports it receives and not the whole structure of the text (i.e. considering then that each supporting claim is completely valid and thus with Cogency 100%). Because of this, it is necessary to update the Major Claim's Cogency value to take into account the context of the essay. Therefore, if $C_{original}$ is the annotated Cogency of the Major Claim, C_i is the annotated Cogency of the supporting claim i and n the number of supporting claims, the new value for the Cogency of the Major Claim is: $C_{new} = C_{original} * (\frac{C_1+...+C_n}{100*n})$
(3) We adopted, for each Major Claim in a text, the following process, divided into three different scenarios:
   (a) When a Major Claim has no counterargument to it presented in the essay. Then $P = C_{new}$.
   (b) When a Major Claim is attacked by a counterargument, which is either refuted by a weak rebuttal or not attacked at all (RC > RR), its quality can only depend on the quality of its supports C_{new}, the strength of the rebuttal (RR) and the weakness of the counterargument (RC). This scenario reflects the rationale that if the counterargument is strong, but the rebuttal is weak, the Major Claim should be weak. Then $P = C_{new} - RC + RR$.
   (c) When a Major Claim is attacked by a counterargument, but the rebuttal is at least as strong or stronger than the counterargument itself (RR $\geq$ RC). In this scenario, given that an argument that successfully anticipates a counterargument and is able to refute it is considered stronger (Wachsmuth, Naderi, et al. 2017), the MDQ of this Major Claim is stronger if the counterargument and the rebuttal also are. Then $P = (RR + RC + C_{new})/3$.
(4) For each Major Claim in a text, we calculated the probability of MDQ as $\mathbb{P}(\text{MDQ}) = \begin{cases} P, & \text{if } 0 \leq P \leq 100; \\ 0, & \text{if } P < 0. \end{cases}$
(5) The final probability $\mathbb{P}$(MDQ) was calculated as the average of its values across all Major Claims in the essay.

For the classification task, labels were assigned using a threshold-based system, with thresholds set at 50% and 70% in distinct scenarios. Essays with probabilities above the selected threshold are labeled as *Good*, while the remaining essays are labeled as *Bad*.

Example 1. *Consider an essay from the AAEC dataset. For its single Major Claim (MC), the annotations provided the following quality scores: Cogency (C) is* 25*, Reasonableness Counterargument (RC) is* 20 *and Reasonableness Rebuttal (RR) is* 10*. Suppose the three supporting claims of the Major Claim have cogencies of* 25*,* 25*, and* 20 *respectively. Thus, the* $\mathbb{P}(MDQ)$ *for this essay is calculated as follows:*

(1) *All metrics are multiplied by four.*
(2) *For the single MC:* $C_{new} = 100 * (\frac{100+100+80}{300}) = 100 * 0.933 = 93$.
(3) *Because RC > RR, we have* $P = 93 - 80 + 40 = 53$.
(4) *Because* $0 \leq P \leq 100$, $\mathbb{P}(\text{MDQ}) = P = 53\%$.
(5) *Because the text has a single Major Claim,* $\mathbb{P}(\text{MDQ}) = 53\%$.

Thus, the essay is 53% effective in supporting its major claim, considering the rebuttal to counterarguments and the quality of its supporting claims. Using a 50% threshold, it is labeled Good, while for a 70% threshold it is labeled Bad.

While generating MDQ scores from the AAEC dataset is valuable, the proposed method employed carries a potential risk of circularity—namely, using derived MDQ scores to validate a system that is itself designed to optimize MDQ. To mitigate this risk and to demonstrate that the system does generalize beyond the data from which the scores were derived, an additional validation step was conducted: a second dataset was employed to

assess whether the system's predictions can approximate broader human perceptions of argumentative quality. The details of this dataset are presented in the next subsection.

2.3.2 ArGPT Dataset. In order to evaluate the Minimal Dialectical Quality and its capacity to generate approximations to human perceptions of argumentation quality, the ArGPT dataset (Rocha, Silveira, et al. 2023) was chosen. This dataset was originally designed to evaluate artificial texts, particularly for identifying contradictions and poorly argued claims. It consists of argumentative essays generated by ChatGPT (version 3.5 that was available during dataset development). The goal was to produce a variety of essays representing strong and weak argumentation.

ArGPT's annotation process was carried out by two annotators experienced in evaluating argumentative essays. It followed a methodology similar to that used for the AAEC dataset. Throughout the process, the annotation guidelines were continuously refined based on discussions and feedback from the annotators.

The second version of ArGPT, which was used in this work, contains 86 themes and 172 essays (with each theme generating two essays). Once generated, the essays were annotated for argumentation mining and quality. Well-established Argument Mining tasks were adopted: span identification, component classification, and relation classification. The argumentative components were annotated as *Major Claims* (central viewpoints) and *Premises* (supporting or attacking other components), while the relationships were classified as *Support* or *Attack*. Using the AM tasks, it was verified that ArGPT's essays closely resemble human argumentation.

To assess the overall quality of the essays, criteria derived from AES, with input from the annotators, were used, including:

- Structure: Evaluation of whether the essay followed an argumentative structure;
- Writing: Assessment of linguistic correctness and the use of argumentative connectives;
- Coherence: Divided into measurements of the essay's adherence to the theme, avoidance of repetition, contradictions, or irrelevant content;
- Truthfulness: Evaluation of the veracity of the presented arguments.

The argumentation quality score itself was derived as the average value of three out of the four subcriteria from coherence (excluding adherence to the essay's theme) and the Truthfulness score. The quality of the generated essays were categorized (using thresholds) into three groups: (1) essays with strong coherence and that argue in support of true claims were labeled as *Good*; (2) essays with flawed argumentation, regardless of whether the defended claim is true or false, were labeled as *Bad*; (3) essays that effectively defend false claims were labeled as *Ugly*.

In the ArGPT dataset, 95 essays were labeled as Bad, 40 as Good, and 37 as Ugly. For the present work, however, the ability to distinguish between true and false claims is not directly relevant, as MDQ focuses primarily on the internal coherence of the argumentation. Accordingly, the following approach was developed to create the argumentation quality metrics that the MIDIQE system is tasked with predicting:

- The criterion about the truthfulness of a claim is excluded, reclassifying all "Ugly" essays as "Good". This adaptation offers a direct way to assess how well MDQ aligns with human annotations of argumentation quality. For the numerical score, we take the original three quality subcriteria: (1) No contradictions found in the essay (C), (2) No repetition of arguments (R), (3) No irrelevant content (I). These three criteria were combined and the final metric computed as $Q = 40 * (C + R + I)/3$, given that each criterion was annotated in a scale from 0 to 2.5.

EXAMPLE 2. *Suppose that scores annotated to each of the three relevant criteria for a given text are* $C = 2$, $R = 2.5$ *and* $I = 2.5$, *and the original label was "Ugly". The adaptation process works as follows: the "Ugly" label simply becomes "Good", reflecting the essay's internal quality of argumentation. The numeric score is calculated as* $Q = 40 * (C + R + I)/3 = 93.33$.

It is important to emphasize that the argumentative quality scores derived from the ArGPT dataset are not intended to represent MDQ itself. Rather, they are human-assigned scores based on distinct criteria explicitly designed to evaluate argument quality. This distinction is precisely what allows these data to serve as a means of validating whether the MDQ and the MDQC effectively capture human judgments of quality. It is also worth noting that the truthfulness criterion is intentionally excluded from this comparison, as MDQ does not consider external factual accuracy, and including it would therefore result in an unfair evaluation.

2.3.3 Evaluation Metrics. MIDIQE has been evaluated using both regression and classification metrics to compare its outputs with the ground truth labels from AAEC and ArGPT. The employed metrics are:

- Mean Squared Error (MSE): Measures the average squared difference between predictions and actual values. The MSE was selected because it is widely used for regression tasks, but it does not account for the ordinal nature of argumentation quality;
- Quadratic Weighted Kappa (QWK): Assesses the agreement between predictions and ground truth while taking into account the ordinal structure of evaluations. The QWK ranges from -1 to 1, where 0 signifies random agreement, 1 indicates perfect agreement, and -1 represents perfect disagreement. This metric is particularly relevant for AES tasks, as maintaining the correct ranking of predictions is as important as minimizing absolute errors. It was chosen as a regression metric, even if its utility is sometimes debated (Yannakoudakis and Cummins 2015), because of AES similarities with argumentation quality;
- F1-Macro: Widely used to assess classification performance, particularly in imbalanced datasets. The F1-Macro calculates the harmonic mean of precision and recall for each class and averages them, ensuring all labels are equally taken into account. It was selected given the unbalanced distribution of labels in both datasets.

3 The Sub-Symbolic Modules

In this section, we present the sub-symbolic and neural models of our MIDIQE system. We begin by introducing our approach to Argumentation Mining in Section 3.1 and move to Single Argument Quality in Section 3.2.

3.1 The Argumentation Mining Module

This section presents the development and evaluation of the Argumentation Mining (AM) module within the MIDIQE system. Subsection 3.1.1 offers a brief overview of the state-of-the-art in AM, while Subsection 3.1.2 delves into the development and performance evaluation of the AM module itself.

3.1.1 Argumentation Mining: Background. AM is an evolving field in Natural Language Processing (NLP) and computational argumentation. It aims to develop techniques to automatically extract arguments, their structure, and their relationships from natural language texts (Lawrence and Reed 2020; Lippi and Torroni 2016). In essence, an AM system takes an argumentative text as input and extracts key argumentative components, constructing an output in the form of an argumentation graph. In this graph, nodes represent argumentative elements, and edges depict the relationships between them.

Over time, several methods have been developed for AM tasks. Early systems primarily relied on hand-crafted features and rules, but these approaches have largely been replaced by machine learning and deep learning methods (Hidayaturrahman et al. 2021). Consequently, numerous datasets have been created, as machine learning models require large quantities of labeled data for training. We later compare several key datasets, as we explore several (deep learning) approaches to implementing each AM task. A significant challenge in AM is the lack of standardization across datasets, with each dataset relying on its own methodology for labeling arguments and organizing argumentative structures. Nevertheless, some similarities can be observed across different datasets, such as in the AAEC and ArGPT discussed before.

A typical AM system involves multiple tasks, not uniformly described across the literature. Several recent proposals define three main tasks that must be addressed in AM (Lawrence and Reed 2020; Morio et al. 2022; Stab and Gurevych 2017). These tasks are:

(1) **Span Identification:** Predicting which parts of the text are argumentative, resulting in a set of extracted components;
(2) **Component Classification:** Categorizing the identified components into predefined labels (e.g., premise, claim);
(3) **Relation Classification:** Determining the relationships (e.g., support, attack) between pairs of components.

Various approaches have been proposed for building AM systems, each targeting specific aspects of the task. These approaches can be broadly categorized into two types: those that address individual AM tasks independently, which are called task-specific in this work, and those that employ a multi-task methodology to handle all tasks within a unified framework.

A notable example of a task-specific approach to AM is IBM's Project Debater. Introduced in 2019 as the first AI system capable of debating human experts on complex topics (Bar-Haim et al. 2021), this system decomposes AM into distinct tasks, utilizing BERT-based classifiers for claim detection, claim boundary identification, and evidence detection, followed by stance classification.

Another task-specific approach was proposed by Lenz et al. (2020). The authors developed a system for extracting argumentative structures as argumentation graphs. Their pipeline first detects and classifies argumentative components as *Claims* and *Premises*. Heuristic methods identify the *Major Claim*, while three graph construction algorithms determine the argumentation structure. Their system demonstrated state-of-the-art component classification performance using random forests and logistic regression.

Several other proposals follow a task-specific approach. In their work on the AbstRCT corpus, Mayer et al. (2020) tackled span identification and relation classification separately using BERT-based models combined with dense neural networks or recurrent networks. Their results showcased the advantages of domain-specific pretraining. Song et al. (2020) introduced Discourse Self-Attention (DiSA), a model that enhances component classification by incorporating positional and relational sentence-level information. Evaluated on multiple datasets, this approach outperformed content-based methods by integrating discourse structure. Hidayaturrahman et al. (2021) demonstrated that Transformer-based architectures improve AM component classification by up to 20% over context-free models such as Word2Vec and GloVe, albeit at the cost of higher computational demands. Guilluy et al. (2023) leveraged constituency trees and Graph Neural Networks (GNNs) for span identification, achieving superior token-level recognition of Argument Discourse Units (ADUs) across datasets such as CDCP. To address data scarcity, Kashefi et al. (2023) introduced ARG-NLI, reframing AM as a Natural Language Inference task, and Bart-PEPT, which utilizes parameter-efficient prompt tuning. Their methods outperformed large language model (LLM) based - zero-shot and fine-tuned - approaches. They also highlighted significant drawbacks of LLMs in AM, including high latency, service reliability issues, and data privacy concerns, underscoring that LLMs are not yet optimal tools for AM.

Concerning the multi-task approach, one early attempt was proposed by Eger et al. (2017). The authors introduced sequence tagging and dependency parsing approaches for joint learning span identification, component classification, and relation classification. Their dependency parsing model, based on a BiLSTM architecture, achieved state-of-the-art performance at the time and showed the value of sharing knowledge across AM tasks.

More recently, Morio et al. (2022) developed a multi-task learning strategy that tackled all AM tasks simultaneously, also evaluating training using multiple corpora. Their system, based on Longformer (Beltagy et al. 2020) and a Span-Biaffine architecture (Dozat and Manning 2018), achieved state-of-the-art results across five datasets, highlighting the benefits of cross-corpus training. Building on this, Huang et al. (2023) introduced

LFAM, which applies BIO-tagging for boundary segmentation, a biaffine dependency parsing structure for relation identification and the maximum tree graph algorithm to improve relation identification. Evaluated on five datasets, LFAM consistently outperformed previous models. Si et al. (2023) focused on the medical domain, proposing Sequential Multi-Task Learning (SeqMT), which integrates Graph Convolutional Networks (GCNs) and domain-specific embeddings (e.g., BioBERT (Lee et al. 2020)). This approach significantly improved AM performance on the AbstRCT dataset. Liu et al. (2023) introduced a multi-turn question-answering framework for AM, using GCNs to refine argument graphs and achieving state-of-the-art results on multiple datasets. Finally, Galassi et al. (2023) proposed a domain-agnostic neural AM model that combines residual networks, multi-task learning, and attention mechanisms. Competing with Transformer-based models like BERT while requiring fewer computational resources, this architecture underscores the potential of scalable multi-task AM systems.

3.1.2 The AM Module. The AM Module in MIDIQE was designed to achieve state-of-the-art performance on argumentative essays, aiming to surpass existing benchmarks whenever possible. To ensure this, all AM iterations were evaluated on the AAEC dataset, which allows direct comparison with prior work. (As ArGPT did not have the required benchmarks, it was only used after the AM Module was finished.)

Given the significant class imbalance inherent in argumentation datasets, the Relation Classification task was divided into two subtasks: (1) Link Detection, which identifies whether a relation exists between two argumentative components, and (2) Relation Classification, which assigns a label to the identified relation.

Following an iterative process inspired by the best results in the argument mining literature, we evaluated multiple approaches until the AM Module reached or surpassed state-of-the-art performance. All experiments were run in an Nvidia DGX A100 system, using a single GPU per experiment. In what follows, we describe the development of each component of the AM Module, while Appendix B presents some unsuccessful – but still informative – attempts, along with additional details on the successful ones.

Span Identification. This AM task is usually modelled as Token Classification. This works as follows: (i) the whole text is tokenized; (ii) the AM model classifies each token either *B* (Begin) - which indicates the beginning of an argumentative component -, *I* (Inside) - indicating the token is part of a component - or *O* (Outside) - which is self-explanatory. This allows for the reconstruction of argument components from the classified tokens.

To create our Span Identification AM module we decided to combine several different AM datasets. Prior research suggests that diverse training data enhances performance across domains (Morio et al. 2022); however, previous approaches have overlooked the similarities between the arguments in all datasets. For the Span Identification task in particular, the annotation methodologies differ very little among AM datasets, as all of them are concerned with finding the argumentative components of a text, even if the framework will eventually classify them differently.

In short, the AM Module for Span Identification was created by training a RoBERTa-Large encoder model on data from different AM datasets combined. Specifically, we combined the following datasets into one single cross-domain Span Identification dataset with 2852 texts:

(1) **ArGPT** (Rocha, Silveira, et al. 2023): This artificial dataset was created using ChatGPT to generate argumentative essays on contradictory themes;
(2) **AAEC** (Stab and Gurevych 2017): Comprising 402 student essays on controversial topics, this dataset distinguishes between argumentative and non-argumentative parts, as noted already in Section 2.3.1;
(3) **MTC** (Peldszus and Stede 2016): This dataset contains 112 micro-texts written by students in response to trigger questions;
(4) **CDCP** (Park and Cardie 2018): This dataset consists of 731 user comments responding to policy proposals related to Consumer Debt Collection Practices (CDCP);

(5) **AbstRCT** (Mayer et al. 2020): Focusing on the medical domain, this dataset contains 669 abstracts from randomized controlled trials (RCTs) sourced from the MEDLINE database[1];
(6) **AASD** (Accuosto and Saggion 2020): This corpus addresses the lack of annotated data for AM in the scientific domain and contains only 60 texts;
(7) **ASRD-Debater** (Shnarch et al. 2020): Part of IBM's Debater Project,[2] this dataset focuses on the identification of argumentative sentences in debates, primarily used for Span Identification. It contains 700 texts, but only those containing arguments were taken, thus reducing the number of examples from 700 to 260.
(8) **Araucaria** (Reed 2006): This dataset consists of argument analyses from various domains, providing both the original texts and their corresponding argumentation trees with annotated relations such as attack and support. It provides 665 texts, but we used only the fully annotated English samples (approximately 67% of the dataset).

The original AAEC-only training set was also tested as a fine-tuning set. Tests with different pretraining and fine-tuning strategies determined that training exclusively on the cross-domain dataset for 100 epochs provided the best results on the target dataset AAEC. All tested models were trained using HuggingFace's token classification framework[3], with standard hyperparameters, with a learning rate of 2×10^{-6} and a batch size of 4. Additional details on our tests are given in Appendix B.6.

Component and Relation Classification. Most AM methods rely on sequence classification, where sentences or sentence pairs are classified independently. This approach, however, ignores contextual dependencies, meaning the same sentence or relation can serve different functions – e.g., as a premise or claim or as an attack or support – depending on its role within the broader argument structure.

To address this limitation, we propose a straightforward approach: we incorporate the full text as input context. For instance, in Component Classification, both the component and full-text context are provided as inputs. Similarly, Relation Classification incorporates the full text along with the pair of components. This insight leverages the known fact that Transformer-based language models are very good in dealing with context in texts to address the mentioned problem.

So, a RoBERTa-Large encoder was employed using the HuggingFace's Sequence Classification framework. Models were trained with a learning rate of 2×10^{-6}, a batch size of 4, and 20 training epochs. Given these settings met performance goals, hyperparameter optimization was left for future work. More details about the implementation can be found in Appendix B.5.

Link Detection. Given that AM systems ultimately construct argumentation graphs, it makes sense to investigate GNN-based approaches to perform such tasks, as this type of neural networks was created to deal with graph-related tasks. Our Link Detection AM Module was built based on this insight. Its model was built using the Deep Graph Library (DGL),[4] incorporating insights from existing Graph Attention Networks[5] (GAT) (**velickovic2018**) and training methodologies for edge classification.[6]

Each argumentative component's text was encoded using RoBERTa-Large embeddings, which were then used as node features in the GNN. The Link Detection Task was modeled as an edge classification task; using GAT architectures with LSTM-based layers for text processing. The model then outputs a label *Link* when there is a relation between the component and *None* otherwise.

[1] https://www.nlm.nih.gov/medline/index.html
[2] https://research.ibm.com/interactive/project-debater/
[3] https://huggingface.co/
[4] https://docs.dgl.ai/en/latest/index.html
[5] https://docs.dgl.ai/en/0.8.x/tutorials/models/1_gnn/9_gat.html
[6] https://docs.dgl.ai/en/latest/guide/training-node.html

Table 1. Comparison with best models in the literature for each AM task.

	Span	Component		Link	Relation	
	F1	F1	Macro	F1	F1	Macro
AM Module	**85.42**	86.41	**84.95**	**70.89**	**69.54**	**62.67**
BLCC	-	-	63.23	-	34.82	-
DiSA	-	-	74.2	-	-	-
MT-All	85.20	**87.68**	80.37	67.88	66.91	55.84
Bart-PEPT	-	-	73	-	-	-
LFAM	85.23	87.08	79.47	68.65	67.49	55.63

Most default DGL parameters were retained, with modifications to the number of hidden layers, learning rates (LR), and training epochs. Learning rates were optimized through a sweep between 0.2 and 2×10^{-6} over 1000 epochs. Stable models with decreasing training loss were prioritized, and among these, the best-performing were selected. Using the chosen LR, the number of epochs was optimized to ensure model convergence. The size of the hidden layers was also adjusted, with explorations ranging from 2 layers to 15 times the maximum size of the tokenized input. The optimal hidden layer size was determined based on training time and performance improvements. Given this, the final training settings were 300 epochs, LR = 2×10^{-3}, hidden layer size = 90. More details can be found in Appendix B.4.

Comparison with the literature. To assess the performance of the AM solutions just outlined and compare them with existing literature, the F1-Micro and Macro scores are reported here, as commonly used in the field (Morio et al. 2022). The AM module was evaluated across four key tasks: (1) Span Identification: The model predicts a set of spans, each represented as a tuple (s, e), where s and e denote the start and end tokens of the span; (2) Component Classification: Identified spans are assigned a label, yielding predictions of the form (s, e, c), where c represents the component type; (3) Link Detection: Relationships between components are predicted as tuples $(s_{src}, e_{src}, s_{tgt}, e_{tgt}, r)$, with s_{src} and e_{src} referring to the source span, s_{tgt} and e_{tgt} to the target span, and r denoting either *Link* or *None*; (4) Relation Classification: Links between components are assigned relation labels, where r represents the specific relation type.

For each AM task, golden outputs are denoted by $\mathcal{G}_{task}$, and system outputs as $\mathcal{S}_{task}$. Precision (P) is defined as $P = |\mathcal{G}_{task} \cap \mathcal{S}_{task}|/|\mathcal{S}_{task}|$, recall ($R$) as $R = |\mathcal{G}_{task} \cap \mathcal{S}_{task}|/|\mathcal{G}_{task}|$, and the F1-score is $F1 = 2PR/P + R$.

Table 1 compares the performance of the built AM Module against state-of-the-art models, including BLCC (Eger et al. 2017), DiSA (Song et al. 2020), MT-All (Morio et al. 2022), Bart-PEPT (Kashefi et al. 2023), and LFAM (Huang et al. 2023). For a fair comparison, results for the AM module were obtained by first applying the Link Detection submodule, followed by relation classification.

The AM module achieves state-of-the-art performance in Span Identification and outperforms existing models in Component Classification (Macro). Additionally, its Link Detection model surpasses prior approaches by over 2%, while the Relation Classification model also yields superior results, which validates the rationale of dividing those tasks. These findings confirm that incorporating full-text context, graph-based features and a cross-domain dataset enables significant performance improvements over the baselines.

These results show that the AM Module reached the goal we set for this module, achieving in essence the state of art on the AAEC dataset. For the ArGPT dataset, as no prior results exist for this dataset, no comparative analysis was conducted.

3.2 Single Argument Quality Assessment

This subsection presents the Single Argument Quality Assessment Module of the MIDIQE system. Because background on Argument Quality Assessment has been given in Section 2, here we just summarize relevant work and provide a detailed account of the module's development.

The SAQ Module serves a specific function within the MIDIQE system, distinct from the system's overall objective. While both address aspects of Argument Quality, the module focuses on a "local" analysis, assessing the relative strength of individual arguments, evaluating whether the premises supporting a conclusion are adequate and whether attacks between arguments are compelling. In contrast, the MIDIQE system processes the outputs of the SAQ Module to evaluate the overall quality of argumentation of a text.

Among the dimensions of Argument Quality outlined previously, the SAQ Module specifically targets logical and dialectical aspects. It evaluates cogency and reasonableness, incorporating the reasonableness counterargument and reasonableness rebuttal metrics as defined by Marro et al. (2022). Consistent with the overall system's design, rhetorical aspects are excluded from the evaluation. An argument is deemed cogent if its premises are both individually acceptable and relevant to its conclusion (Marro et al. 2022; Wachsmuth, Naderi, et al. 2017). Reasonableness, on the other hand, depends on the quality of its counterarguments and rebuttals, reflecting the principle that argumentation is reasonable when it meaningfully contributes to resolving an issue (Wachsmuth, Naderi, et al. 2017).

We compare our approach with that of the original work by Marro et al. (2022). For their model designed to predict the Cogency metric, the authors evaluated several approaches, including Random Forests, SVMs, and LSTM-based architectures. As input features, they combined text embeddings generated by a Longformer encoder with graph embeddings produced by FEATHER-G. Their best performance was obtained by feeding both types of embeddings into an SVM classifier, with the second-best results achieved by a Random Forest. When relying solely on linguistic features, the Random Forest model performed best. Note however that, for the Reasonableness metric, the authors reported that the dataset was too small to support training machine learning models. Consequently, they developed heuristic rules based on the Cogency scores predicted by their Cogency models.

As this work was the first to be conducted on the dataset, it was not compared against other baselines; however, it serves as the reference baseline for the purposes of our SAQ module.

3.2.1 The SAQ Module Models. To develop our SAQ Module, we utilized the same SAQ annotations from the AAEC dataset provided by Marro et al. (2022). These annotations were used to train and test the model architectures and were provided on a five-point scale: $[0, 10, 15, 20, 25]$, where 0 represents the lowest quality and 25 the highest. The Reasonableness metric was further divided into Counterargument and Rebuttal dimensions, though a single model was used for both, as in Marro et al. (2022).

Our SAQ Module was implemented using machine learning models to predict both quality metrics within argumentation graphs. To allow for that, even in a scenario where the availability of data is small, we developed a specialized preprocessing strategy to implicitly encode the structure of the argumentation graph. On the one hand, the model predicting Cogency receives as input the tokens of the target component along with those of its supporting components. On the other hand, the model predicting Reasonableness incorporates tokens from the target component, its attackers, and the supporting components of those attackers. The underlying rationale is that these quality metrics depend on the broader argumentative context surrounding the target component rather than on the target alone.

Two distinct architectures were tested. Both utilized embeddings generated by a RoBERTa Large model and were adapted for specific quality dimensions:

Table 2. Performance comparison of BiLSTM and Random Forest for 5, 3 and 2 different labels to predict the quality labels. Available baseline values for 3 labels are also shown. Numbers are Macro scores.

	Cogency	Reasonableness
BiLSTM - 5 labels	29.52	13.45
Random Forest - 5 labels	41.69	33.31
BiLSTM - 3 labels	44.60	31.18
Random Forest - 3 labels	72.26	66.02
Best Result (Marro et al. 2022)	77	54
Second Best Result (Marro et al. 2022)	75	18
Best Linguistic Result (Marro et al. 2022)	74	-
BiLSTM - 2 labels	74.15	42.81
Random Forest - 2 labels	89.75	92.69

(1) A Random Forest model, directly inspired by Marro et al. (2022), implemented using the scikit-learn[7] library. A class weight array was applied to address class imbalances.
(2) A BiLSTM-based model with variations tailored to each metric. For implementation, the Keras framework[8] was used with default hyperparameters, training with a batch size of 32 for 30 epochs.

3.2.2 SAQ Module Tests and Comparison with Baselines. Models were tested using five, three, and two-label variations. Class imbalance was minimized using only three labels, so labels 10 and 15 were combined, as were labels 20 and 25. Furthermore, we also tested a two-label solution by combining labels 0 and 10, as well as labels 15, 20, and 25.

Table 2 reports results averaged across ten random seeds; for the three-label setting, baseline performances from the literature are also included. For the Cogency metric, the Random Forest model consistently outperforms the BiLSTM across all label configurations. As expected, reducing the number of labels leads to higher performance. In the three-label setting, the Random Forest-based SAQ Cogency module also achieves results comparable to the state-of-the-art, supporting its inclusion in the MIDIQE pipeline.

A similar pattern is observed for Reasonableness. Once again, the Random Forest model outperforms the BiLSTM, and performance increases as the number of labels decreases. In this case, however, the Random Forest model achieves the best overall results, surpassing the reported baselines by approximately 12%. These results provide additional support for the effectiveness of the proposed preprocessing strategy.

Given these results, the two Random Forest models were selected to predict the Cogency and Reasonableness metrics, respectively. Further optimization of the BiLSTM architecture was left for future work, as the Random Forest models were sufficient for the purposes of this paper. Readers interested in the BiLSTM architecture are referred to Appendix C, which provides additional implementation details.

Overall, the proposed SAQ module achieves performance comparable to the state-of-the-art for Cogency and surpasses existing approaches for Reasonableness, while remaining sufficiently effective for integration into the MIDIQE framework.

3.2.3 Post-processing of Quality Labels. The SAQ Module outputs quality metrics in their original form or combined-label format. Because the rest of the MIDIQE system operates using probabilities, these labels were converted into interpretable probabilities by multiplying each metric by 4, thus placing them within a $[0, 100]$

[7]https://scikit-learn.org
[8]https://keras.io/

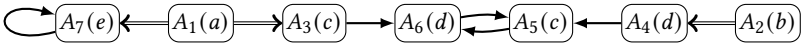

Fig. 3. A Bipolar Conclusion-augmented Argumentation Graph. Single-edge relations are representations of attacks and double-edged ones are supports. A_3 stands for an argument that defends conclusion c. A conclusion can be defended by different arguments.

scale. Consequently, Cogency is taken as the probability that an argument's supports are collectively effective, while Reasonableness is taken as the probability that an argument's attacks are effective.

4 Symbolic Module

This section introduces the symbolic part of MiDiQE, i.e. its Argumentation Reasoning module. It briefly reviews relevant literature approaches, then presents its conceptual foundations, and details the module's development. Readers interested in more detail about the theoretical background for our AR Module are referred to Appendix D.

4.1 Argumentation Reasoning: Theoretical Foundations

The AR module builds upon two complementary recent developments in computational argumentation: the Bipolar Conclusion-Augmented Argumentation Framework (BCAF) (Rocha and Cozman 2022b) and the L-credal semantics (Rocha and Cozman 2022a). In this section, we provide a brief summary of those contributions and the reasoning for choosing them, but we refer to the original published papers for more details.

The first of those developments, the BCAFs, represent an important generalization of the classical Abstract Argumentation Frameworks (AAFs) proposed by Dung (1995) and of the Bipolar Argumentation Frameworks (BAFs) introduced by Cayrol and Lagasquie-Schiex (2013). AAFs determine acceptable argument sets by analyzing attack relations among abstract arguments, whereas BAFs enrich this framework with support relations, enabling a representation of reasoning that more closely aligns with human patterns. Meanwhile, BCAFs incorporate the advantages of those models, but it also explicitly links each argument to the claims or conclusions they justify, making the reasoning process slightly less abstract.

This explicit linkage between arguments and conclusions provides a key theoretical advantage: it enables a direct correspondence between argumentation frameworks and logic programs, allowing algorithms and semantics from logic programming to be applied seamlessly to argumentation. More specifically, BCAFs achieve a formal one-to-one equivalence between argumentation semantics and logic programming semantics, specifically linking the semi-stable semantics in argumentation to the L-stable semantics in logic programming. This result resolves a long-standing mismatch between the two fields, ensuring that equivalent reasoning produces equivalent results regardless of the formalism used. And more than semantics equivalence, using the modified WGC algorithm, it is possible to translate a BCAF into a logic program and vice-versa without any informational loss.

The following example (extracted from (Rocha and Cozman 2022b)) illustrates the one-to-one equivalence between BCAFs and logic programs.

Example 3. *Consider the following logic program:*

$$c \text{ :- } a., \qquad d \text{ :- } b., \qquad c \text{ :- } \textbf{not } d., \qquad d \text{ :- } \textbf{not } c.,$$

$$e \text{ :- } a, \textbf{not } e., \qquad a., \qquad b..$$

Using translations in the literature (Rocha and Cozman 2022b), we can translate this program to obtain the equivalent argumentation graph shown in Figure 11. At the same time, we can translate the graph back to the original program. If we apply the semi-stable BCAF argumentation semantics on the graph or if we apply the L-stable

Table 3. Left: tight probability intervals for Example 4; if an interval contains a single number, the interval is replaced by the number. Right: probabilities using the SMProbLog approach by Totis et al. (2021).

	v = true	v = false	v = undefined
$\mathbb{P}(a = v)$	0.5	0.5	0.0
$\mathbb{P}(b = v)$	0.5	0.5	0.0
$\mathbb{P}(c = v)$	[0.5, 0.75]	[0.25, 0.5]	0.0
$\mathbb{P}(d = v)$	[0.5, 0.75]	[0.25, 0.5]	0.0
$\mathbb{P}(e = v)$	0.0	0.5	0.5

	v = true	v = false	v = inconsistent
$\mathbb{P}(a = v)$	0.0	0.5	0.5
$\mathbb{P}(b = v)$	0.25	0.25	0.5
$\mathbb{P}(c = v)$	0.125	0.375	0.5
$\mathbb{P}(d = v)$	0.375	0.125	0.5
$\mathbb{P}(e = v)$	0.0	0.5	0.5

semantics to the logic program, we obtain the same results: all conclusions are considered acceptable, except for e, which is considered undecided.

On the probabilistic side, the L-credal semantics extends the BCAF foundation to handle uncertainty in argumentative inference. Several previous proposals integrate probabilities into logic programming but rely on the stable semantics, which can yield multiple or even no models. To manage this issue, those approaches often assume a probability distribution over models, which then introduces unintended biases and fails to capture different kinds of uncertainties in the argumentation process. The L-stable semantics, by itself, addresses the case where no model would be produced by the stable semantics by selecting the models with the minimal number of undecided conclusions. In argumentation scenarios it is not satisfactory to abort reasoning just because the chosen semantics is unable to produce a model and the L-stable semantics guarantees a solution.

To deal with the multiple models scenario, the L-stable semantics is expanded, creating the L-credal semantics. This means that reasoning is performed over sets of probability distributions instead of fixed ones, which allows for a non-biased approach to the distribution of the probability mass among models and also enables the representation of different sources of uncertainty.

The following example (extracted from (Rocha and Cozman 2022a)) illustrates the difference in reasoning of a more traditional solution to a probabilistic logic program and the one executed by the L-credal semantics, highlighting the latter's advantages.

Example 4. *Assume the following probabilistic logic program:*

$$c \text{ :- } a.; \qquad d \text{ :- } b., \qquad c \text{ :- } \textbf{not } d., \qquad d \text{ :- } \textbf{not } c.,$$

$$e \text{ :- } a, \textbf{not } e., \qquad 0.5 :: a., \qquad 0.5 :: b..$$

There are four total choices – i.e. one per probabilistic fact where we choose whether the atom is in the program or not – ($\emptyset$, $\{a\}$, $\{b\}$, $\{a, b\}$), each with probability 0.25. *For total choice $\emptyset$, the induced logic program has two L-stable models (both are also stable models): one where every atom is* false *except c that is* true*, and another where every atom is* false *except d that is* true. *For total choice $\{a\}$, there is a single L-stable model (but no stable model): e is* undefined*, while both a and c are* true *and both b and d are* false. *For total choice $\{b\}$, there is again a single L-stable model (that is the single stable model): a, c and e are* false *and both b and d are* true. *Finally, for total choice $\{a, b\}$ there is a single L-stable model (but again no stable model): all atoms are* true *except e that is* undefined. *By adding probabilities across models for all possible distributions in the semantics, the tight probability intervals in Table 3 are obtained (left).*

It is interesting to compare the L-stable semantics results with those obtained by the SMProbLog approach by Totis et al. (2021), where probabilities are distributed uniformly between models and the non-existence of stable models leads to a probability mass being assigned to inconsistent *values. Table 3 (right) shows the probabilities obtained via the SMProbLog approach. Notably, the probabilities of* inconsistent *values are remarkably high, even for atoms that can always be assigned other values.*

It is also relevant to note that the L-credal semantics nicely differentiates between: (i) the uncertainty captured by the probabilities in probabilistic facts; (ii) the uncertainty regarding multiple stable (l-stable) models, captured by probability intervals; (iii) the uncertainty arising from inability to satisfy constraints, leading to non-zero probability for undefined *values. This cannot be done using the SMProbLog approach, as we can see it mixed (ii) and (iii) in the same* inconsistent *value.*

Together, BCAFs and the L-credal semantics provide a robust theoretical foundation for our AR module. BCAFs supply the structured, interpretable representation of arguments and their bipolar relations, while the L-credal semantics supplies the probabilistic reasoning layer that quantifies uncertainty and belief. In addition, BCAFs allow for established logic programming algorithms to be used to solve argumentation problems, as discussed in the next section.

4.2 Argumentation Reasoning Module: Implementation

Using the theoretical foundations of reasoning overviewed in the previous section, the AR Module was built by integrating the BCAF formalism with the probabilistic reasoning allowed by the L-credal semantics. The idea is to translate the argumentation graphs produced by the AM and SAQ Modules, which are annotated with probabilistic relationships between arguments, into probabilistic BCAFs. Multiple strategies might be considered for this, given the BCAF's abstract treatment of what is an argument. The present work adopts the most straightforward approach: we treat each component extracted from the text as an independent argument with its own conclusion.

After this step, the BCAF representation is translated into a probabilistic logic program using the adapted WCG algorithm. At this point, however, an additional adjustment is required to properly encode the uncertainty information produced by the SAQ module.

The key issue is that the SAQ module does not assign probabilities to individual argumentative relations. Instead, for each argument, it predicts uncertainty values collectively for each type of relation—for example, a single probability representing the overall quality of all support relations an argument receives, and another single probability representing the overall quality of its attack relations. In other words, supports are evaluated jointly, as are attacks, rather than on a relation-by-relation basis.

By contrast, the logic program obtained directly from the BCAF treats each support and each attack separately within a single rule. To illustrate this mismatch, consider an argument whose conclusion is a, which is supported by two arguments with conclusions b and c, and attacked by two arguments with conclusions d and e. In the original BCAF-based logic program, this argument is represented by the rule

$$r : a \text{ :- } b, c, \textbf{not } d, \textbf{not } e.$$

This formulation implicitly assumes that the influence of each support and each attack is handled individually. However, since the SAQ module provides only one aggregated quality value for all supports and one for all attacks, this rule cannot directly incorporate the predicted uncertainty information.

For this reason, the rule must be rewritten into an equivalent form that explicitly reflects the aggregated treatment of supports and attacks, allowing the probabilistic information produced by the SAQ module to be correctly integrated. Thus, the equivalent rules are:

$$\begin{aligned} r1 &: acceptable(a) \text{ :- } supports(a), \textbf{not } attacks(a). \\ r2 &: supports(a) \text{ :- } supQuality(a), acceptable(b), acceptable(c). \\ r3 &: attacks(a) \text{ :- } atkQuality(a), \textbf{not } failedAttacks(a). \\ r4 &: failedAttacks(a) \text{ :- } \textbf{not } acceptable(d), \textbf{not } acceptable(e). \end{aligned}$$

Fig. 4. Example of interpretability of the system's outputs for an Argumentation Graph. MC: Major Claim; C: Claim; P: Premise.

Assuming both the quality metrics $supQuality(a)$ and $atkQuality(a)$ to be certain, one can easily see how both representations are equivalent. However, the second representation can convey the combined quality of supports and attacks. This process is repeated for all rules in the original BCAF program.

To run this probabilistic logic program and get the acceptability results for each Major Claim in the text (needed to estimate $\mathbb{P}(MDQ)$), we have employed the dPASP package (Geh et al. 2023), as it can handle probabilistic logic programs using the credal L-stable semantics. Given this, dPASP computes the probability that each Major Claim in an essay is accepted. However, because MDQ requires all Major Claims to be accepted, we construct a second logic program that builds on the results of the first, as illustrated in the example below:

Example 5. *Consider an essay containing two identified Major Claims. The program below evaluates the justification of each Major Claim—that is, the combination of its acceptability and the probability that the component is indeed a Major Claim—and then integrates these results to compute the probability that the essay qualifies as a strong argumentative text according to MDQ:*

$$
\begin{aligned}
MDQ &:\!- \; justified(A_0), justified(A_1),\\
justified(A_0) &:\!- \; majorclaim(A_0), acceptable(A_0),\\
justified(A_1) &:\!- \; majorclaim(A_1), acceptable(A_1),\\
0.8 :: majorclaim(A_0), &\qquad 0.9 :: majorclaim(A_1),\\
0.9 :: acceptable(A_0), &\qquad 0.8 :: acceptable(A_1).
\end{aligned}
$$

Upon execution of the logic program, the probability that the essay meets the minimum dialectical quality test is determined as $\mathbb{P}(MDQ) = 51.84\%$.

That is, running the logic program yields a probabilistic evaluation of the essay's ability to justify its main points. Moreover, the system's outputs are inherently interpretable, as demonstrated by the following example:

Example 6. *Consider the argumentation graph extracted from an essay shown in Figure 4. Suppose the system has determined that the essay is of poor quality. This outcome can be easily understood by observing the graph: the Major Claim is effectively attacked by Claim C2, while its only support, Claim C1, is defeated by Premise P1.*

To summarize, the AR Module bridges argumentation and probabilistic logic programming, employing the dPASP package to handle the L-credal semantics and the input BCAFs. By applying probabilistic reasoning, the module produces interpretable probabilistic assessments of MDQ using the information provided by MIDIQE's sub-symbolic models.

Regarding running time performance, even the longest logic program (the argumentation graph with more nodes and relations) needed no more than a few seconds to provide its answers. Given this and the justifications provided earlier, we believe our AR Module does produce the reasoning performance we desired.

5 The Complete MIDIQE System: Results

The previous two sections introduced the components of the MIDIQE system, namely its sub-symbolic and symbolic modules. The neural AM Module extracts the argumentative structure of an essay, while the SAQ Module evaluates the quality of the relations between the extracted arguments. By integrating this information,

Table 4. Testing the AR Module of the MIDIQE system on the AAEC dataset, using perfect AM and SAQ Modules. Macro (50/70): Macro score for a threshold of 50%/70%.

	Macro (50)	Macro (70)	MSE	QWK
No SAQ	68.33	58.61	1990.91	0.48
SAQ Full (Original)	**73.13**	66.45	**991.04**	0.66
SAQ Full (3 labels)	69.94	**68.20**	1058.06	**0.67**
SAQ Full (2 labels)	72.11	62.38	1555.47	0.60
SAQ Support (Original)	67.93	64.25	1511.31	0.45
SAQ Support (3 labels)	70.03	67.57	1537.98	0.49
SAQ Support (2 labels)	66.44	63.74	2071.92	0.41
SAQ Attack (Original)	70.58	56.40	1532.94	0.63
SAQ Attack (3 labels)	71.93	56.40	1557.93	0.61
SAQ Attack (2 labels)	66.17	49.29	1894.71	0.56

the symbolic AR Module determines the acceptability levels of the text's main points and assesses the probability of its Minimal Dialectical Quality. This section presents an evaluation of the complete MIDIQE pipeline.

As discussed previously, the MIDIQE pipeline is tested in two distinct ways. The first evaluation validates whether the system accurately assesses the MDQ and the MDQC. This requires comparing its outputs with annotations explicitly provided for MDQ in the AAEC dataset. The second evaluation assesses whether computing the probability of MDQ yields results that align with human perceptions of argumentation quality. To this end, the ArGPT dataset, which has quality annotations based on human criteria independent of MDQ, was used.

5.1 First Evaluation: Validating the MIDIQE System

In Section 2.3, original human annotations for Cogency and Reasonableness for each essay from the AAEC dataset were combined into a proxy for MDQ. Comparing the output of the MIDIQE system with those ground-truth annotations provides a solid basis for assessing MIDIQE's ability to capture the MDQ. This comparison is made using the metrics detailed before, i.e. F1-Macro for classification and MSE and QWK for regression.

To enrich the evaluation of the system's capabilities, its results were compared with baseline values. Most of the baseline results were generated by fine-tuning several commonly used encoder models to predict MDQ labels, as well as to function as regression models for predicting MDQ probabilities. They were fine-tuned for 10 different seeds, using a learning rate of 2×10^{-6}, a batch size of 8, with 20 epochs. For the regression models, the learning rate was 2×10^{-5} with 40 epochs.

Additionally, the baselines for the classification tasks include two of the most prominent and widely discussed large language models (LLMs) available nowadays: OpenAI's GPT-4.0[9] and DeepSeek-R1.[10] API resources for both models were used with different temperature settings to generate results. To maintain consistency, both LLMs were given the same prompt, which was carefully refined through an extensive trial-and-error process. This prompt, available in Appendix E, provided a detailed explanation of MDQ, its relevance within the literature, and illustrative examples of argumentative essay quality. For the regression tasks, the use of those models is somewhat ambiguous, as explaining the scores in natural language seemed arbitrary; for this reason, no prompt was created.

The experiments on the AAEC dataset were run incrementally, as in an ablation study, adding the system's modules one at a time to assess their individual impact on the final results. First, the AR Module was tested

[9]https://openai.com/index/chatgpt/
[10]https://www.deepseek.com/

Table 5. Performance of the AR Module on the AAEC dataset, using perfect AM and SAQ Modules, compared with baseline values. Macro (50/70): Macro score for a threshold of 50%/70%.

	Macro (50)	Macro (70)	MSE	QWK
RoBERTa-Base	47.42 ± 7.07	46.80 ± 5.31	828.58 ± 187.75	0.21 ± 0.10
RoBERTa-Large	44.22 ± 5.49	51.07 ± 5.66	731.89 ± 187.04	0.12 ± 0.18
Longformer-Base	49.63 ± 5.35	44.04 ± 5.0.5	648.55 ± 140.05	0.21 ± 0.06
Longformer-Large	45.17 ± 5.90	54.76 ± 8.32	822.16 ± 151.50	0.22 ± 0.12
DeBERTa	40.56 ± 8.54	50.73 ± 8.52	651.31 ± 172.28	0.15 ± 0.13
GPT 4.0	35.12 ± 10.89	22.13 ± 7.03	-	-
Deepseek-R1	49.94 ± 2.40	32.73 ± 1.92	-	-
SAQ Full (5 labels)	**73.13**	66.45	911.04	0.65
SAQ Full (3 labels)	69.94	**68.20**	1058.06	**0.67**

using ground truth annotations for both AM and SAQ. Then, the SAQ Module was introduced while still relying on ground truth for AM. Finally, the AM Module was fully deployed, enabling an evaluation of the complete MiDiQE system.

5.1.1 Testing the AR Module. In order to assess the AR Module's effectiveness in evaluating MDQ independently of the other modules, it was necessary to assume perfect inputs for AM and SAQ. Therefore, *ground truth values for both modules* were used in these tests. However, because the SAQ values were preprocessed into 5, 3, and 2 label versions and considering that the quality of supports and attacks may differently impact the AR Module's performance, all configurations of SAQ ground truth labels were tested. Table 4 presents the results for all possible combinations, including the absence of SAQ values.

The results show that incorporating the SAQ module does not significantly degrade performance in the worst-case scenario and, in several configurations, leads to measurable improvements. However, the impact of the SAQ support component depends strongly on the number of labels used in the classification setup.

Across most evaluation metrics, the configuration with three labels yields the best overall performance, followed by the five-label configuration. Notably, the three-label setting is the only one that consistently improves all reported metrics simultaneously. In contrast, the five-label setting improves roughly half of the metrics while negatively affecting the remaining ones. The two-label configuration shows the weakest impact, yielding an improvement only in the Macro (70).

These results suggest that a moderate level of label granularity strikes the most effective balance for exploiting the information provided by the SAQ support module.

The trend on the number of labels is the similar for the SAQ attack approach. However, in this case the Macro (50) usually benefits more, while the Macro (70) metric is impacted negatively. In all cases, the regression metrics improve when compared to the no SAQ results.

Finally, the combination of SAQ support and attack values produces the best results. The results for 5 and 3 labels are the best for half of the metrics each, with improvements of about 5% for the Macro (50) score and almost 10% to the Macro (70) when compared to the no SAQ approach. Similarly, the regression metrics all improve significantly.

Given these findings, using the combined SAQ Module to evaluate support and attack relations with 5 or 3 labels provides the most reliable performances in practice. Thus, with the use of the SAQ Module fixed, it is possible to evaluate whether the AR Module, using ground truth AM and SAQ, can accurately assess the Minimal Dialectical Quality of essays. In Table 5, its performance is compared with baseline values.

Table 6. Testing the MIDIQE system, featuring both ground truth and predicted AM with predicted SAQ outputs, on the AAEC dataset. Macro (50/70): Macro score for a threshold of 50%/70%.

Perfect AM and Predicted SAQ				
	Macro (50)	Macro (70)	MSE	QWK
No SAQ	**68.33**	58.61	1990.91	0.48
SAQ Full (Original)	65.73	60.78	1636.68	**0.58**
SAQ Full (3 labels)	68.17	**62.95**	**1572.69**	0.56
SAQ Full (2 labels)	61.97	57.01	1998.50	0.47
SAQ Support (Original)	65.47	59.56	2012.55	0.44
SAQ Support (3 labels)	65.47	59.56	2021.71	0.44
SAQ Support (2 labels)	62.91	56.77	2235.22	0.38
SAQ Attack (Original)	63.86	51.71	1983.37	0.52
SAQ Attack (3 labels)	66.83	53.83	1833.88	0.55
SAQ Attack (2 labels)	65.27	52.89	1882.05	0.56
Predicted AM and SAQ				
	Macro (50)	Macro (70)	MSE	QWK
No SAQ	68.33	50.60	2473.19	0.31
SAQ Full (Original)	65.49	55.90	2131.56	0.24
SAQ Full (3 labels)	**70.80**	**57.07**	**1785.91**	0.32
SAQ Full (2 labels)	67	53.06	2575.72	0.20
SAQ Support (Original)	68.44	55.53	2058.26	0.27
SAQ Support (3 labels)	70.44	55.53	1957.82	0.31
SAQ Support (2 labels)	68.89	53.67	2461.78	0.22
SAQ Attack (Original)	65.62	52.06	2451.48	0.31
SAQ Attack (3 labels)	69.58	54.26	2225.05	**0.35**
SAQ Attack (2 labels)	65.62	49.83	2588.38	0.32

Firstly, based on the Macro metrics for the AR Module, with both close to or exceeding 70, it is reasonable to conclude that it is indeed capable of evaluating the quality of argumentation in essays using MDQ when the outputs of AM and SAQ are perfect. This conclusion becomes even more evident when these metrics are compared to the baseline values, as the AR Module outperforms all of them, except for the MSE metric. A second point arises when comparing the module's outputs with the variation in the baseline results. The best baseline results for classification metrics, achieved by Longformer-Large, exhibit a standard deviation of about 8. This indicates that the baseline models are slightly unreliable due to their sensitivity to the chosen seed. In contrast, the AR Module presented here consistently produces the same results and also holds the advantage of providing interpretable results, as was previously demonstrated.

Regarding the regression metrics, the QWK metric suggests that the probability scores assigned by the AR Module align closely with the distribution of the ground truth scores. Interestingly, however, the module produces a slightly higher MSE than the baseline values. This implies that while it better captures the overall distribution of the ground truth - i.e. its ordinal order -, the AR Module's individual predictions may deviate more from the actual values. Considering the earlier emphasis on the need for a more user-friendly metric to evaluate argumentation quality, a higher QWK score seems more significant than a lower MSE. This perspective is further illustrated in the following example.

Table 7. Comparison of the MiDiQE system's performance with baseline values on the AAEC dataset.

	Macro (50)	Macro (70)	MSE	QWK
RoBERTa-Base	47.42 ± 7.07	46.80 ± 5.31	828.58 ± 187.75	0.21 ± 0.10
RoBERTa-Large	44.22 ± 5.49	51.07 ± 5.66	731.89 ± 187.04	0.12 ± 0.18
Longformer-Base	49.63 ± 5.35	44.04 ± 5.0.5	648.55 ± 140.05	0.21 ± 0.06
Longformer-Large	45.17 ± 5.90	54.76 ± 8.32	822.16 ± 151.50	0.22 ± 0.12
DeBERTa	40.56 ± 8.54	50.73 ± 8.52	651.31 ± 172.28	0.15 ± 0.13
GPT 4.0	35.12 ± 10.89	22.13 ± 7.03	-	-
Deepseek-R1	49.94 ± 2.40	32.73 ± 1.92	-	-
Perfect AM and SAQ	**73.13**	**66.45**	911.04	**0.65**
Perfect AM	68.17	62.95	1572.69	0.56
MiDiQE I (SAQ Full 3 labels)	70.80	57.07	1785.91	0.32
MiDiQE II (SAQ Attack 3 labels)	69.58	54.26	2225.05	0.35

Example 7. *Consider a dataset containing only two essays, A and B, with ground truth probabilities of* 60 *and* 70*, respectively. Two regression models, I and II, are evaluated. Model I produces predictions of* 50 *for A and* 90 *for B, resulting in an MSE of* 250 *and a QWK of* 0.6*. Model II, on the other hand, predicts* 65 *for A and* 60 *for B, achieving a much lower MSE of* 62.5 *but a negative QWK of* −0.67*. Despite its higher MSE, Model I produces better results because it correctly identifies that essay B is better than essay A, as indicated by the ground truth. In contrast, Model II, with its lower MSE, incorrectly suggests the opposite, misrepresenting the relative quality of the essays, which is more valuable for a human than the actual probabilistic values.*

Therefore, based on the results obtained, it can be concluded that, when using ground truth AM and SAQ values, the AR Module successfully predicts the Minimal Dialectical Quality of argumentation, outperforming baseline models.

5.1.2 Testing the Sub-symbolic Modules. The next step in MiDiQE's validation is to evaluate the influence of the sub-symbolic models. Our first set of experiments focused on the SAQ Module while assuming ground truth data for AM. The next set involved assessing the influence of the AM Module, thereby enabling the evaluation of the complete MiDiQE system. The results of this comprehensive evaluation, again with different configuration of the SAQ Module, are presented in Table 6.

Similar to the results obtained with ground-truth SAQ values, the best performance is achieved when using the complete SAQ Module with three labels, both when combined with ground-truth AM outputs and when paired with the values predicted by the AM Module. The use of the SAQ Module leads to a noticeable drop in performance relative to the ground-truth setting. In particular, the predicted Cogency scores, while still beneficial in some cases, tend to hurt the results the most in others. Another observation is that the performance gap between the different SAQ configurations becomes less pronounced. In fact, in the fully predicted setting (both AM and SAQ), the configuration using only SAQ attack with three labels even surpasses the complete SAQ Module in terms of QWK.

Interestingly however, the upper part of Table 6 also reveals that, despite a clear performance decline relative to Table 4, the best configurations in both tables remain fairly comparable. This suggests that, in the best-case scenario, using predicted SAQ values can approximate the effectiveness of using ground-truth values.

In addition, a comparison of the two steps in Table 6 shows a drop across all metrics—except for Macro (50), where results improve slightly when relying on predicted AM and SAQ outputs. This behavior appears across almost all SAQ configurations. In a complex system such as MiDiQE, errors in one module (e.g., AM)

Table 8. Comparison of the MIDIQE system on ArGPT using different SAQ configurations. Macro (50/70): Macro score for a threshold of 50%/70%.

	Macro (50)	Macro (70)	MSE	QWK
No SAQ	67.61	67.61	2100.00	0.25
SAQ Full (Original)	80.57	73.33	1995.73	0.29
SAQ Full (3 labels)	65.37	65.37	2244.21	0.17
SAQ Full (2 labels)	67.61	67.61	2402.08	0.20
SAQ Support (Original)	70.91	70.91	2504.65	0.23
SAQ Support (3 labels)	56.36	56.36	2916.08	0.07
SAQ Support (2 labels)	56.36	56.36	3100.0	0.10
SAQ Attack (Original)	75.00	68.63	1645.83	0.27
SAQ Attack (3 labels)	**81.18**	**81.18**	**1355.21**	**0.45**
SAQ Attack (2 labels)	75.00	75.00	1402.08	0.37

can propagate and distort predictions in another (e.g., SAQ). Occasionally, however, those interacting errors can produce outputs that align better with a specific evaluation threshold — even if overall performance degrades. In this case, the combined imperfections lead the system to assign slightly higher scores to essays, which improves performance for the arbitrary threshold of 50 while harming it for the remaining metrics. As expected, using the predicted AM Module instead of ground-truth labels negatively impacts performance in nearly all configurations, with only this single exception.

Taking into account all these (nontrivially interacting) observations, the configuration using the complete SAQ Module with three labels remains the best choice across both settings examined in Table 6. However, this configuration is closely followed by the configuration with SAQ attack and 3 labels. Therefore, we consider both feasible configurations to be adopted in the full MIDIQE system.

5.1.3 Testing MIDIQE. With the influence of each module analyzed, the performance of both configurations for the complete system can now be compared to baseline values. Table 7 presents the results of these experiments, also illustrating the incremental impact of each module on the performance of MIDIQE.

As expected, the performance using ground truth values for both AM and SAQ achieves the highest scores for most metrics, with the exception of MSE (even though it is well within the baseline's significant intervals). Introducing the predicted SAQ and subsequently the predicted AM leads to performance declines; however, compared to the baseline values, the complete MIDIQE system still performs admirably well. For the Macro (50) metric, it still outperforms all baselines and for the Macro (70) metric configuration I outperforms them all, while configuration II ranks slightly below the average performance of the Longformer-Large model but falls well within its margin of error (8.32). The same can be said about MIDIQE's performance when compared to the GPT and Deepseek baselines.

Although the MSE metric for both MIDIQE configurations remains higher than the baseline values, they outperform all average baseline values in the QWK metric. These results demonstrate that the complete system effectively and reliably predicts the minimal dialectical quality of essays. With the completion of the experiments on the AAEC dataset, the first phase of evaluations is concluded. The results demonstrate that MIDIQE effectively predicts MDQ, and outputs usually exceed, or at least rival, the possible alternatives we took as baselines, while also offering interpretable outputs.

Table 9. Comparison of the MIDIQE's performance with baseline values on ArGPT

	Macro (50)	Macro (70)	MSE	QWK
RoBERTa-Base	50.74 ± 13.62	50.74 ± 13.62	**404.12 ± 88.89**	**0.49 ± 0.13**
RoBERTa-Large	68.57 ± 7.70	68.57 ± 7.70	482.60 ± 171.37	0.28 ± 0.20
Longformer-Base	51.21 ± 5.76	51.21 ± 5.76	508.88 ± 173.38	0.27 ± 0.21
Longformer-Large	64.45 ± 12.22	64.45 ± 12.22	535.73 ± 114.80	0.21 ± 0.20
DeBERTa	62.73 ± 9.82	62.73 ± 9.82	460.28 ± 66.30	0.26 ± 0.28
GPT 4.0	62.06 ± 5.78	62.06 ± 5.78	-	-
Deepseek-R1	72.32 ± 4.30	72.32 ± 4.30	-	-
MIDIQE	**81.18**	**81.18**	1355.21	**0.45**

5.2 Validating the Minimal Dialectical Quality Criterion

Having demonstrated that the MIDIQE system can successfully predict MDQ, it is crucial to evaluate whether MDQ, particularly as it is predicted by the system, is suitable as a measure of overall argumentative quality in essays as perceived by humans. To address this, the ArGPT dataset was employed, as it was annotated for quality based on criteria distinct from MDQ.

As described before, to handle the ground truth quality values of the ArGPT dataset, we adapted the original annotations by removing the factual validation criterion, which resulted in reclassifying all "Ugly" labels as "Good". For regression metrics, this approach added the values of the three remaining criteria – excluding factual validity – used to generate the original labels.

As with the analysis of the AAEC dataset, the evaluation begins by assessing the impact of different submodules of SAQ and label granularities, followed by a comparison with the baseline models. Table 8 presents the impact of the SAQ submodules on the system results.

Firstly, it is important to note that the ArGPT dataset is unfortunately small, which does lead to some repeated values in the Table even under different configurations. Secondly, the results indicate that the SAQ support submodule generally has a negative effect on final performance when compared to variants of the system without SAQ. The only exception is the five-label configuration, in which support information yields improvements for the classification metrics, though not for the regression ones. In contrast, the SAQ attack submodule consistently improves performance across all label granularities, producing the strongest overall results — especially in the three-label setting. The combination of both SAQ submodules yields mixed outcomes, except in the case of 5 labels granularity.

Taking into account these findings, and recalling that ArGPT serves as our "test dataset", we selected among the two top-performing MIDIQE configurations identified on the AAEC dataset using their performance on the ArGPT dataset. Accordingly, the final configuration adopted for MIDIQE is the one using only the SAQ–attack submodule with three labels of granularity, as it is among the best performers for MDQ prediction on AAEC and the most effective at applying MDQ to predict overall quality (as perceived by humans) on ArGPT.

With the best SAQ Module combination defined for MIDIQE, Table 9 compares its performance with baseline values. It is clear that MIDIQE achieved strong results when compared to the baseline models. It outperformed all other values on the classification metrics and achieved solid results on the regression metrics. For the latter, MIDIQE tied with RoBERTa-Base for the best performance on the QWK metric, taking into account the latter's significant variations. As expected, the baseline values outperformed both system configurations for the MSE, however, again, the good performance on the QWK metric indicates MIDIQE functions as good regression models.

Therefore, the results in Table 9 show that the MIDIQE system can effectively approximate the quality of argumentative essays as humans perceive it, even as it focuses only on MDQ, rather than the broader set of quality

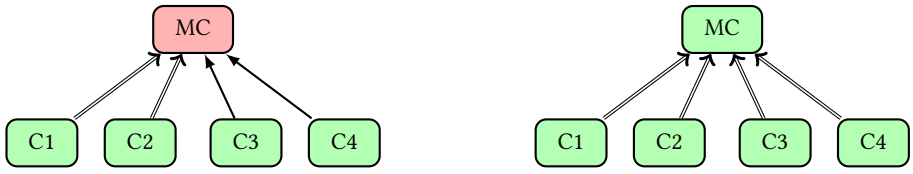

Fig. 5. Argumentation Graph extracted from AAEC. Left: Ground-truth graph; Right: Graph predicted by the AM Module wrongly labeling all relations as supports and leading to the incorrect acceptance of MC. MC: Major Claim; C: Claim.

dimensions presumably used by human annotators. It not only outperformed the baseline models across multiple metrics, and produced competitive results for the rest, but also offered interpretable results. This validates the MIDIQE system, along with MDQ and the MDQC, as valuable tools for automatic evaluation of argumentative quality in essays.

5.3 Error Analysis and Discussion

Even though the MIDIQE system was shown to predict MDQ and to be aligned with human evaluation of quality in essays, it does make mistakes. Some of those mistakes, in particular when evaluating the ArGPT dataset, can be attributed to the fact that the annotation criteria do not involve the MDQ directly. There are, however, other types of mistakes produced by MIDIQE, even when it is predicting MDQ on the AAEC dataset.

The various errors were classified into five types, as we went through them. The first type originates from the AM Module, where incorrect labeling of relations, missing arguments, wrongly predicted links, and similar issues may lead to flawed argumentation graphs, which later result in inaccurate reasoning. The second type of error involves SAQ Module's assessments. For instance, an attack on a Major Claim may be mistakenly considered bad, leading to the incorrect acceptance of that Major Claim. The third type of error happens because MIDIQE (in SAQ attack only) assumes that all support relations are perfect and this may fail. The fourth type of error originates from some combination of the first three.

The fifth and most interesting type of error occurs when the system produces an incorrect answer despite performing all its steps correctly. While this situation suggests a limitation in MIDIQE, it also illuminates one of its main features: its inherent interpretability. By examining the argumentation graph and the system's response, one can often find a reasonable explanation for the error. This makes the system useful even when it errs, as it can serve as a valuable secondary "opinion" in situations such as annotation scenarios, helping annotators refine their subjective judgments, or in educational scenarios where students are practicing the writing of essays.

And because argument annotation is a *really* tricky and difficult task, as discussed in the literature, we would like to emphasize that such interpretability is a major positive aspect of our overall approach.

In this remainder of this section, examples of each type of error are provided, highlighting that even when the system errs, it retains its interpretability. For all the argumentation graphs provided, an edge with a single line must be interpreted as an attack, and a double line as a support. Additionally, all ground truth labels are produced with a threshold of 50%, even though similar results can be reached using a threshold of 70%.

5.3.1 Argumentation Mining Error. Consider the following subset of components extracted from an essay on the AAEC dataset and the respective ground truth argumentation graph on the left side of Figure 5. To make the example easier to understand, the number of components in the graph is restricted to those that are relevant to the analysis, excluding the rest:

- MC: government should decide for punishment on a case by case basis;
- C1: if there will be a fixed punishment for each type of crime then it will lead to increase the crime;
- C2: it is necessary to have a look behind crime;

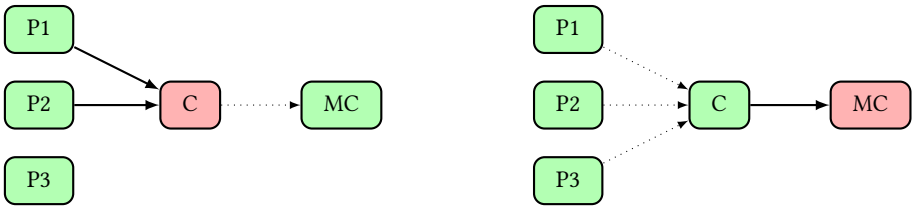

Fig. 6. Partial Argumentation Graph extracted from AAEC. Left: Ground truth graph with strong attacks (full line) from $P1$ and $P2$ to C and a weak attack (dotted line) from C to MC. Right: Predicted graph with the wrong attack from $P3$ to C and the attacks from the premises to C being wrongly predicted as weak and the attack from C to MC wrongly predicted as strong. MC: Major Claim; C: Claim; P: Premise.

- C3: fixed punishment will reduce the time taken by judges to give their judgment and it will be effective economically for the government;
- C4: fixed punishment will reduce the gap between poor and rich people and everyone will understand the importance of each other.

Looking at the graph on the left of Figure 5, and knowing that the excluded parts do not influence it, it can be seen that because $C3$ and $C4$ are attacking the Major Claim and are not rebutted, the Major Claim is defeated, thus leaving the text with the label *Bad*. This is indeed the ground truth label for this essay. However, the AM Module, instead of predicting the attacks from $C3$ and $C4$ to MC, labeled those relations as supports (right on Figure 5). Given this AM mistake, there is now no direct attack on the Major Claim, making it acceptable and leading the MIDIQE system to incorrectly label the essay as *Good*.

5.3.2 Quality of Single Argument Error. In the example provided in this section, the error of the SAQ Module is determinant in leading the system to label the text wrongly. However, there is also an AM mistake. Thus, this result serves as an example both of the error derived from the SAQ error and as an error resulting from a combination of factors.

Consider the following subset of argumentative components extracted from an AAEC essay and the partial argumentation graph constructed by the AM Module in Figure 6. As before, only the components that are relevant to the example are shown.

- MC: children should learn music and art in a young age;
- C: parents shouldn't put so much stress on children;
- P1: it depends on how you value these learning activities to be like;
- P2: it will not be such pressure on children;
- P3: for a child, music and art are more like an open door to introduce the art world than training area to acquire specific skills.

If only the argumentation graph, without taking any quality metric into account, is taken into account, one might think that the Major Claim is defended from its attacker C by the rebuttals P. The ground truth quality labels for the relations, represented on the left side of Figure 6, express that the attack of C on MC is weak (dotted line), while the attacks C receives are strong (full line). These ground truth labels indicate that the first attack failed and the others were successful, leading the system to reject C and accept MC, thus labeling the essay as *Good*. This is indeed the ground truth label for this essay.

The MIDIQE system, however, made two mistakes when evaluating this essay, as shown on the right side of Figure 6. First, according to the ground truth of AM, there is no relation between $P3$ and C; that is, the attack from $P3$ to C observed in the graph does not exist. Even though this is an AM mistake, it alone would not change

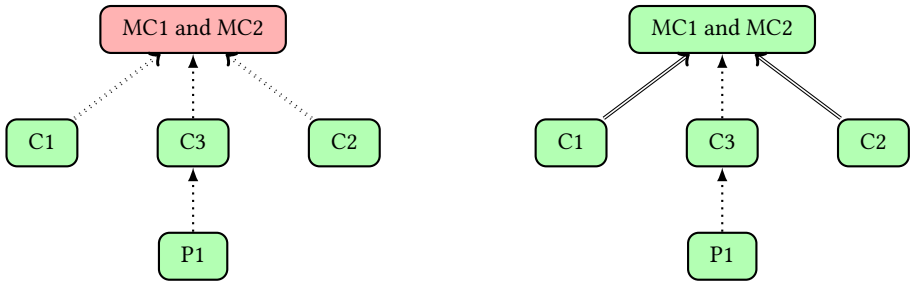

Fig. 7. Argumentation Graph extracted from AAEC. Left: Ground truth where the supports MC receives are considered weak (dotted). Right: Because the MIDIQE configuration does not use the SAQ submodule for support, all supports are deemed perfect (full line). MC: Major Claim; C: Claim; P: Premise.

the system's labeling of the essay, because the attacks from $P1$ and $P2$ on C are still valid. However, the second mistake leads to the wrong answer. The SAQ Module incorrectly predicts that the attack from C to MC is strong, while the attacks C receives are weak. This causes the pipeline to accept C and reject MC, thus wrongly labeling the essay as *Bad*.

5.3.3 Error Due to the Quality Metric for the Support Relation. As explained earlier, the chosen configuration for the MIDIQE system does not make use of the SAQ submodule responsible for predicting the quality of supports. However, in some cases, not using such a model produces incorrect results. This sometimes happens because annotators may consider an essay bad because the reasons given for believing the Major Claims are weak, and, while the system performs all the steps correctly, it fails to capture this nuance. This section exemplifies such an error by analyzing an essay extracted from the AAEC dataset. Consider the following argumentative components and the argumentation graph in Figure 7:

- MC1: the benefits of children engagement in paid work outweigh its drawbacks;
- MC2: if children take part in some kind of paid work with their parents' permission, they can learn lively lessons and become mature and responsible;
- C1: they are likely to understand their parents' hardship, to respect values of money and to form a good habit of saving;
- C2: when children take jobs, they tend to be more responsible;
- C3: children will be more material, neglect their study for earning money or be exploited by the employers;
- P1: if children get good care and instructions from their parents, they can take advantages of the work to learn valuable things and avoid going in a wrong way.

The depicted argumentation graphs correspond to both the ground truth as well as the predicted graph by the AM Module (left and right sides of Figure 7), so there is no AM mistake. Both SAQ ground truth labels and the SAQ Module predictions state that the attacks of $P6$ to $C3$ and from $C3$ to the Major Claims are weak, making the latter acceptable. However, the annotators judged that the reasons $C1$ and $C2$ given to believe in the Major Claims are weak (dotted), thus making $MC1$ and $MC2$ unsupported and unacceptable. The ground truth label for this text is *Bad*, but because the MIDIQE system always assumes the support relations are perfect, it is unable to identify the nuance provided by the annotators, thus leading it to the mistake.

5.3.4 Error with the Whole Pipeline Working Perfectly. Sometimes, an error the system occurs even though all its modules work perfectly. To illustrate that, an example is taken where both AM and SAQ Modules output their ground truth values. Consider the following argumentative components and the corresponding argumentation graph in Figure 8:

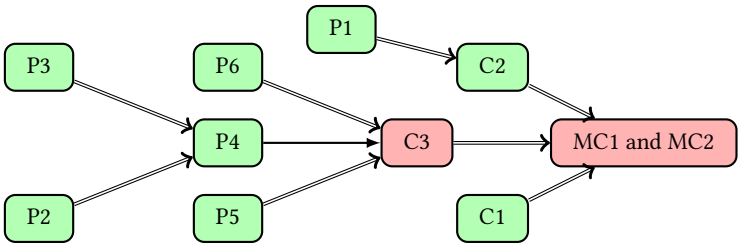

Fig. 8. Argumentation Graph illustrating the MiDiQE's reasoning to assess the essay as "Bad". Despite all MiDiQE's modules working perfectly, the ground truth label for this essay was "Good". MC: Major Claim; C: Claim; P: Premise.

- MC1: it is time for the government to take action to control the proliferation of violent scenes in media;
- MC2: what the government should do immediately is to strengthen censorship to control the amount of violence in media;
- C1: our society is preoccupied with violent scenes;
- C2: this will definitely set a bad example for the immature audiences who, lacking correct parental guidance and abilities to differentiate the right from wrong, are prone to go astray by imitating the violent behaviors and committing real violent crimes;
- C3: excessive violence would offer misleading information to the public and exert detrimental influence to the society as well;
- P1: they regard the violence as reasonable and justifiable;
- P2: media, such as TV or film, is considered as the correct information source regulated by the government;
- P3: it should be reporting and reflecting true phenomena in society;
- P4: violence, as one of the facts in the real world, certainly has to be reported;
- P5: it has been recently reported in the news that a seventeen-year-old boy killed all his family members, one sister and two parents out of hatred and jealousy;
- P6: in the real world, we are inclined to take extreme approaches to tackle a small problem due to excessive violence exposure.

The MiDiQE system labels this essay as *Bad*, since $P4$ successfully attacks $C3$, a necessary claim for the acceptance of $MC1$ and $MC2$. While the system correctly follows all steps, utilizing both AM and SAQ ground truths, and the reasoning process is accurate, the actual ground truth label for this essay is *Good*.

When assessing the quality of the argument relations, the annotators judged that although $C3$ is successfully attacked, it did not significantly undermine the overall quality of the essay. This highlights some important limitations of the MiDiQE system. Specifically, this demonstrates that the proposed approach to evaluating text quality through the concept of MDQ has its shortcomings, as it can still err even when functioning correctly. Here, the argumentative structure clearly suggests the label *Bad*, which the system produces, but the human annotators who analyzed the essay considered the flaw in reasoning to be a minor one.

Given the existing literature, the system's limitation in assessing quality beyond what is implicit in the argumentative structure is understandable. It does not account for other well-established dimensions of argument quality, such as the three dialectical and three rhetorical dimensions proposed by Wachsmuth, Naderi, et al. (2017). It only evaluates the minimal dialectical dimension and a more comprehensive evaluation of argumentation quality would likely require integrating at least some of these additional dimensions.

Despite its errors, the system provides an interpretable analysis, as shown in Figure 8, highlighting a key feature of the proposed approach. It is reasonable to imagine that an annotator using the MiDiQE system to assist in the annotation process might view the system's output as a different, yet valid, perspective, potentially

prompting a re-evaluation of the essay. In this case, the system places significant emphasis on $P4$, suggesting that its mere inclusion completely weakens the writer's argument. This could offer valuable insight to an annotator – especially considering the subjective nature of the annotation process – by encouraging a different consideration of the argument's strength.

6 Conclusions and Future Work

This paper introduced the Minimal Dialectical Quality dimension, a novel approach to assessing argumentative quality that is defined by the ability of an argumentation to justify and defend its own main conclusions using only the information explicitly present in the text. Positioned within the existing literature on argumentation quality, MDQ does not rely on external knowledge or fact-checking, thus providing a less subjective basis for quality assessment.

Building on this, we proposed the Minimal Dialectical Quality Evaluator, a neuro-symbolic computational system designed to automatically assess MDQ probabilities in argumentative essays and then classify their quality based on comparisons with predefined thresholds. It was developed by integrating the three primary research areas of computational argumentation: Argumentation Mining, Argumentation Quality, and Argumentation Reasoning. This integration by itself already represents a novelty of this work, as prior studies typically combined only two of these areas – for example, Cerveira do Amaral et al. (2023)'s combination of AM and AR – or focused exclusively on one. Additionally, the developed system offers interpretable results, a desirable feature in modern AI systems.

Each stage of the system's development also contributed independently to addressing specific challenges in computational argumentation. For AM, this paper investigated current limitations in the field, proposed a unified cross-domain dataset for Span Identification and demonstrated that this AM task benefited significantly from training on datasets spanning multiple domains. For Link Detection a high performance was achieved when employing Graph Neural Networks, leveraging the information that the output should be an argumentation graph structure as an assumption. Finally, Component and Relation Classification produced the best results when the model was provided with contextual information, incorporating the full text in the input.

For the SAQ Module, we introduced a new preprocessing strategy: to support predictions of Cogency and Reasonableness, the text of the relevant graph nodes was combined into a single input sequence, thereby embedding local argumentation-graph structure directly into the model's input. This approach not only produced substantial performance gains over the baseline for the Reasonableness metric, but also enabled the use of machine learning methods for this metric -— something not explored in prior work. For the Cogency metric, however, performance remained slightly below the baseline.

The AR Module integrates two recent proposals from the literature: the Bipolar Conclusion-Augmented Argumentation Framework, which enables one-to-one translations between argumentation frameworks and logic programming, and L-Credal semantics, which is well-suited for reasoning in uncertain argumentation settings. The implementation of the AR Module and its integration of results from all of MiDiQE's components is also a secondary contribution of this work.

The MiDiQE system was empirically validated in its ability to assess the Minimal Dialectical Quality of argumentative essays using the AAEC dataset. The influence of each module was analyzed, along with the impact of using predicted values versus ground-truth values. Furthermore, through experiments on the ArGPT dataset, both MDQ and the MDQC were shown to produce good approximations of human judgments of argument quality, demonstrating the their potential both as valuable and less subjective metrics for broader argument evaluation, and as potential computational tools to produce that kind of assessment – a current limitation of other metrics found in the literature.

Finally, the MiDiQE system's ability to produce interpretable outcomes emerges as a key advantage. Even when the system errs, its structured reasoning allows users to identify the mistakes, such as incorrect relation classification by the AM Module, or to consider its assessment as a secondary perspective on argumentative quality. This feature holds practical implications for applications such as text annotation and essay writing, where the system could assist users in refining their evaluations and improving their work.

Future research should address the limitations of the MiDiQE system. Although the system has been empirically validated and the Minimal Dialectical Quality criterion shows a strong alignment with human perceptions of argumentative quality, both are deliberately limited in scope. As its name suggests, MDQ captures a minimal dimension of argumentation quality: while it provides a robust and interpretable approximation, it does not aim to cover all aspects of argumentative quality discussed in the literature.

Moreover, by design, MiDiQE relies exclusively on information explicitly present in the text. As a consequence, it does not account for external knowledge, such as the factual correctness of claims or domain-specific background information, which may be crucial for a more comprehensive quality assessment. Extending MiDiQE to incorporate additional quality dimensions and to integrate external knowledge sources therefore constitutes a promising and natural direction for future work.

Other ways to enhance the MiDiQE system are relatively straightforward. One potential improvement involves refining how the argumentation graph, extracted by the AM Module, is translated into the BCAF formalism. In this paper, each identified argumentative component was treated as an independent argument with its own conclusion. While this approach was sufficient to produce well-formed and non-redundant BCAFs, and ultimately yielded good results, it certainly can be expanded. A more sophisticated approach could define arguments as combinations of claims and their premises, aligning more closely with certain definitions of an argument found in the literature. This adjustment would lead to a different SAQ Module and reduce the number of arguments that the AR Module needs to process, potentially improving overall system performance.

Beyond conceptual improvements, enhancing each sub-symbolic module individually could further improve results, as demonstrated by the performance gap between using predicted values and ground-truth annotations for AM and SAQ. For the AM Module, for instance, a direct way to build on the insights produced by this paper would be to combine the most effective ideas tested. For instance, results showed that neural models performed better when they had access to the full essay context during classification. Additionally, training with the cross-domain minimalist dataset generally improved performance. Integrating both of these insights into a single approach for tasks could lead to even better results for tasks such as Component Classification, Relation Classification, and potentially Link Detection.

For the SAQ Module, a natural future direction is to combine the strengths of our preprocessing strategy with the graph embeddings used in the baseline method, potentially leading to substantial performance gains. In addition, exploring architectures that integrate an encoder head with a dedicated classification head may further enhance predictive accuracy.

Finally, to fully explore the potential of the MiDiQE system, it should be tested on a broader range of text types beyond argumentative essays. This requires developing and annotating datasets for other genres of text, a nontrivial and costly task.

7 Code Availability

The source code for MiDiQE and all our experiments is freely available at https://github.com/victorhnr/MiDiQE.

Acknowledgments

This work was carried out in part at the Center for Artificial Intelligence (C4AI-USP), with support by the São Paulo Research Foundation (FAPESP grant 2019/07665-4) and IBM Corporation. V. H. N. Rocha was partially supported by

the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) grants 88887.616392/2021-00 and 88887.916704/2023-00. F. G. Cozman was partially supported by CNPq grants 312180/2018-7 and 305753/2022-3. We acknowledge support by CAPES - Finance Code 001. Serena Villata was supported by the French government, through the 3IA Cote d'Azur Investments in the project managed by the National Research Agency (ANR) with the reference number ANR-23-IACL-0001.

We also would like thank Ryan Riegel, Achille Fokoue-Nkoutche, Alexander Gray, and Tian Gao from the IBM Corporation; Theo Kollias, Mariana Chaves, Cristian Cardellino, Nicolás Benjamin Ocampo and Greta Damo from the Université Côte d'Azur Research Team; and Prof. Denis Deratani Mauá, Prof. Glauber de Bona, Igor Silveira, Paulo Pirozelli, Renato Geh, Adriano Augusto Antongiovanni, Yago Primerano Arouca and Prof. Sarajane M. Peres from the C4AI for their valuable contributions to this project.

References

P. Accuosto and H. Saggion. 2020. "Mining arguments in scientific abstracts with discourse-level embeddings." *Data Knowledge Engineering*, 129, 101840. DOI: https://doi.org/10.1016/j.datak.2020.101840.

G. Alfano, S. Greco, F. Parisi, and I. Trubitsyna. Sept. 2020. "On the Semantics of Abstract Argumentation Frameworks: A Logic Programming Approach." *Theory and Practice of Logic Programming*, 20, 5, (Sept. 2020), 703–718. DOI: 10.1017/S1471068420000253.

T. Augustin, F. P. A. Coolen, G. de Cooman, and M. C. M. Troffaes. 2014. *Introduction to Imprecise Probabilities*. Wiley.

D. Bahdanau, K. Cho, and Y. Bengio. 2015. "Neural Machine Translation by Jointly Learning to Align and Translate." In: *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*. https://arxiv.org/abs/1409.0473.

R. Bar-Haim, Y. Kantor, E. Venezian, Y. Katz, and N. Slonim. Nov. 2021. "Project Debater APIs: Decomposing the AI Grand Challenge." In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, (Nov. 2021), 267–274. DOI: 10.18653/v1/2021.emnlp-demo.31.

C. Baral, M. Gelfond, and N. Rushton. 2009. "Probabilistic Reasoning with Answer Sets." *Theory and Practice of Logic Programming*, 9, 1, 57–144.

P. Baroni, M. Caminada, and M. Giacomin. 2004. "An introduction to argumentation semantics." *The Knowledge Engineering Review*, 1–24.

I. Beltagy, M. E. Peters, and A. Cohan. 2020. "Longformer: The Long-Document Transformer." *CoRR*, abs/2004.05150. arXiv: 2004.05150.

T. B. Brown et al.. 2020. "Language Models are Few-Shot Learners." In: *Proceedings of the 34th International Conference on Neural Information Processing Systems* (NIPS '20) Article 159. Curran Associates Inc., Vancouver, BC, Canada, 25 pages. ISBN: 9781713829546.

E. Cabrio and S. Villata. July 2012. "Combining Textual Entailment and Argumentation Theory for Supporting Online Debates Interactions." In: *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Ed. by H. Li, C.-Y. Lin, M. Osborne, G. G. Lee, and J. C. Park. Association for Computational Linguistics, Jeju Island, Korea, (July 2012), 208–212.

M. Caminada, S. Sá, J. Alcântara, and W. Dvorak. 2015. "On the equivalence between logic programming semantics and argumentation semantics." *International Journal of Approximate Reasoning*, 58, 87–111.

C. Cayrol and M.-C. Lagasquie-Schiex. 2013. "Bipolarity in argumentation graphs: Towards a better understanding." *International Journal of Approximate Reasoning*, 54, 7, 876–899. Special issue: Uncertainty in Artificial Intelligence and Databases.

F. Cerveira do Amaral, H. S. Pinto, and B. Martins. 2023. "Argumentation Mining from Textual Documents Combining Deep Learning and Reasoning." In: *Progress in Artificial Intelligence*. Ed. by N. Moniz, Z. Vale, J. Cascalho, C. Silva, and R. Sebastião. Springer Nature Switzerland, Cham, 389–401. ISBN: 978-3-031-49008-8.

O. Cocarascu, A. Rago, and F. Toni. 2019. "From formal argumentation to conversational systems."

F. G. Cozman and D. D. Mauá. 2017. "On the semantics and complexity of probabilistic logic programs." *Journal of Artificial Intelligence Research*, 60, 221–262.

J. Devlin, M.-W. Chang, K. Lee, and K. 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 (NAACL)*, 4171–4186.

T. Dozat and C. D. Manning. July 2018. "Simpler but More Accurate Semantic Dependency Parsing." In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Ed. by I. Gurevych and Y. Miyao. Association for Computational Linguistics, Melbourne, Australia, (July 2018), 484–490.

P. M. Dung. 1995. "On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games." *Artificial Intelligence*, 77, 2, 321–357.

W. Dvorák, A. Rapberger, and S. Woltran. 2020a. "Argumentation semantics under a claim-centric view: Properties, expressiveness and relation to SETAFs." *17th International Conference on Principles of Knowledge Representation and Reasoning, KR 2020*, 1, 340–349. ISBN: 9781713825982. DOI: 10.24963/KR.2020/35.

W. Dvorák, A. Rapberger, and S. Woltran. Aug. 2020b. "On the relation between claim-augmented argumentation frameworks and collective attacks." In: *Frontiers in Artificial Intelligence and Applications*. Vol. 325. IOS Press BV, (Aug. 2020), 721–728. ISBN: 9781643681009. DOI: 10.3233/FAIA200159.

W. Dvorák and S. Woltran. 2019. "Complexity of abstract argumentation under a claim-centric view." *33rd AAAI Conference on Artificial Intelligence, AAAI 2019, 31st Innovative Applications of Artificial Intelligence Conference, IAAI 2019 and the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019*, 2801–2808. ISBN: 9781577358091. DOI: 10.1609/AAAI.V33I01.33012801.

F. H. van Eemeren. 2018. *Argumentation Theory: A Pragma-Dialectical Perspective*. Argumentation Library. Springer International Publishing. ISBN: 978-3-319-95381-6. DOI: https://doi.org/10.1007/978-3-319-95381-6.

S. Eger, J. Daxenberger, and I. Gurevych. July 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)*. Association for Computational Linguistics, Vancouver, Canada, (July 2017), 11–22. DOI: 10.18653/v1/P17-1002.

R. El Baff, H. Wachsmuth, K. Al-Khatib, and B. Stein. Oct. 2018. "Challenge or Empower: Revisiting Argumentation Quality in a News Editorial Corpus." In: *Proceedings of the 22nd Conference on Computational Natural Language Learning*. Ed. by A. Korhonen and I. Titov. Association for Computational Linguistics, Brussels, Belgium, (Oct. 2018), 454–464. DOI: 10.18653/v1/K18-1044.

D. Fierens, G. Van den Broeck, J. Renkens, D. Shrerionov, B. Gutmann, G. Janssens, and L. De Raedt. 2014. "Inference and learning in probabilistic logic programs using weighted Boolean formulas." *Theory and Practice of Logic Programming*, 15, 3, 358–401.

A. Galassi, M. Lippi, and P. Torroni. 2023. "Multi-Task Attentive Residual Networks for Argument Mining." *IEEE/ACM Transactions on Audio Speech and Language Processing*, 31, 1877–1892. eprint: 2102.12227. DOI: 10.1109/TASLP.2023.3275040.

R. L. Geh, J. Gonçalves, I. C. Silveira, D. D. Mauá, and F. G. Cozman. 2023. *dPASP: A Comprehensive Differentiable Probabilistic Answer Set Programming Environment For Neurosymbolic Learning and Reasoning*. (2023). arXiv: 2308.02944 (cs.AI).

S. Guilluy, F. Mehats, and B. Chouli. 2023. "Constituency Tree Representation for Argument Unit Recognition." In: *Proceedings of the 10th Workshop on Argument Mining*. Association for Computational Linguistics, Stroudsburg, PA, USA, 35–44. ISBN: 9798891760509. DOI: 10.18653/v1/2023.argmining-1.4.

Hidayaturrahman, E. Dave, D. Suhartono, and A. M. Arymurthy. Dec. 2021. "Enhancing argumentation component classification using contextual language model." *Journal of Big Data*, 8, 1, (Dec. 2021), 103. DOI: 10.1186/s40537-021-00490-2.

J. Huang, Q. Fang, S. Wang, Z. Tian, F. Liu, Z. Huang, and D. Li. 2023. "Argumentative Relationship Recognition Based on End-to-End Multitask Learning." *Proceedings of ACM Conference (Conference'17)*, 1.

A. Hunter and K. Noor. Apr. 2020. "Aggregation of Perspectives Using the Constellations Approach to Probabilistic Argumentation." *Proceedings of the AAAI Conference on Artificial Intelligence*, 34, 03, (Apr. 2020), 2846–2853. DOI: 10.1609/aaai.v34i03.5674.

A. Hunter, S. Polberg, and M. Thimm. Apr. 2020. "Epistemic graphs for representing and reasoning with positive and negative influences of arguments." *Artificial Intelligence*, 281, (Apr. 2020), 103236. DOI: 10.1016/J.ARTINT.2020.103236.

O. Kashefi, S. Chan, and S. Somasundaran. Dec. 2023. "Argument Detection in Student Essays under Resource Constraints." In: *Proceedings of the 10th Workshop on Argument Mining*. Ed. by M. Alshomary, C.-C. Chen, S. Muresan, J. Park, and J. Romberg. Association for Computational Linguistics, Singapore, (Dec. 2023), 64–75. DOI: 10.18653/v1/2023.argmining-1.7.

T. N. Kipf and M. Welling. 2017. "Semi-Supervised Classification with Graph Convolutional Networks." In: *Proceedings of the 5th International Conference on Learning Representations (ICLR)*.

T. Kuribayashi, H. Ouchi, N. Inoue, J. Suzuki, P. Reisert, T. Miyoshi, and K. Inui. Dec. 2020. "An Empirical Study of Span Representations in Argumentation Structure Parsing." *Journal of Natural Language Processing*, 27, 4, (Dec. 2020), 753–779. ISBN: 9781950737482. DOI: 10.5715/jnlp.27.753.

P. Lagakis and S. Demetriadis. 2021. "Automated essay scoring: A review of the field." In: *2021 International Conference on Computer, Information and Telecommunication Systems (CITS)*, 1–6. DOI: 10.1109/CITS52676.2021.9618476.

G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer. 2016. "Neural Architectures for Named Entity Recognition." In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 260–270.

J. Lawrence and C. Reed. Jan. 2020. "Argument Mining: A Survey." *Computational Linguistics*, 45, 4, (Jan. 2020), 765–818. ISBN: 9781608459858.

J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang. Feb. 2020. "BioBERT: a pre-trained biomedical language representation model for biomedical text mining." *Bioinformatics*, 36, 4, (Feb. 2020), 1234–1240. Ed. by J. Wren. eprint: 1901.08746. DOI: 10.1093/bioinformatics/btz682.

M. Lenz, P. Sahitaj, S. Kallenberg, C. Coors, L. Dumani, R. Schenkel, and R. Bergmann. Aug. 2020. "Towards an argument mining pipeline transforming texts to argument graphs." *Frontiers in Artificial Intelligence and Applications*, 326, (Aug. 2020), 263–270. ISBN: 9781643681061. eprint: 2006.04562. DOI: 10.3233/FAIA200510.

M. Lippi and P. Torroni. Mar. 2016. "Argumentation Mining: State of the Art and Emerging Trends." *ACM Trans. Internet Technol.*, 16, 2, Article 10, (Mar. 2016), 25 pages. DOI: 10.1145/2850417.

B. Liu, V. Schlegel, P. Thompson, T. Batista-Navarro, and S. Ananiadou. 2023. "Global information-aware argument mining based on a top-down multi-turn QA model." *Information Processing and Management*, 60, 103445. DOI: 10.1016/j.ipm.2023.103445.

T. Lukasiewicz. July 2005. "Probabilistic Description Logic Programs." In: *European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2005)*. Springer, Barcelona, Spain, (July 2005), 737–749.

S. Marro, E. Cabrio, and S. Villata. Dec. 2022. "Graph Embeddings for Argumentation Quality Assessment." In: *Findings of the Association for Computational Linguistics: EMNLP 2022*. Ed. by Y. Goldberg, Z. Kozareva, and Y. Zhang. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, (Dec. 2022), 4154–4164. DOI: 10.18653/v1/2022.findings-emnlp.306.

T. Mayer, E. Cabrio, and S. Villata. Aug. 2020. "Transformer-based argument mining for healthcare applications." In: *Frontiers in Artificial Intelligence and Applications*. Vol. 325. IOS Press BV, (Aug. 2020), 2108–2115. ISBN: 9781643681009. DOI: 10.3233/FAIA200334.

G. Morio, H. Ozaki, T. Morishita, and K. Yanai. May 2022. "End-to-end Argument Mining with Cross-corpora Multi-task Learning." *Transactions of the Association for Computational Linguistics*, 10, (May 2022), 639–658. DOI: 10.1162/tacl_a_00481.

D. J. O'Keefe and S. Jackson. 1995. "Argument quality and persuasive effects: A review of current approaches." In: *Argumentation and values: Proceedings of the ninth Alta conference on argumentation*, 88–92.

J. Park and C. Cardie. May 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)*. European Language Resources Association (ELRA), Miyazaki, Japan, (May 2018).

A. Peldszus and M. Stede. 2016. "An Annotated Corpus of Argumentative Microtexts." *Proceedings of the 1st European Conference on Argumentation (ECA'16)*, 801–816.

I. Persing and V. Ng. July 2015. "Modeling Argument Strength in Student Essays." In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Ed. by C. Zong and M. Strube. Association for Computational Linguistics, Beijing, China, (July 2015), 543–552. DOI: 10.3115/v1/P15-1053.

S. Polberg and A. Hunter. 2018. "Empirical evaluation of abstract argumentation: Supporting the need for bipolar and probabilistic approaches." *International Journal of Approximate Reasoning*, 93, 487–543. DOI: https://doi.org/10.1016/j.ijar.2017.11.009.

C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Jan. 2020. "Exploring the limits of transfer learning with a unified text-to-text transformer." *J. Mach. Learn. Res.*, 21, 1, Article 140, (Jan. 2020), 67 pages.

A. Rapberger. 2020. "Defining argumentation semantics under a claim-centric view." In: *CEUR Workshop Proceedings*. Vol. 2655. CEUR-WS.

C. Reed. 2006. "Preliminary Results from an Argument Corpus." In: *Linguistics in the twenty-first century*, 185–196.

R. Rinott, L. Dankin, C. Alzate Perez, M. M. Khapra, E. Aharoni, and N. Slonim. Sept. 2015. "Show Me Your Evidence - an Automatic Method for Context Dependent Evidence Detection." In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Ed. by L. Màrquez, C. Callison-Burch, and J. Su. Association for Computational Linguistics, Lisbon, Portugal, (Sept. 2015), 440–450. DOI: 10.18653/v1/D15-1050.

V. H. N. Rocha and F. G. Cozman. 2022a. "A Credal Least Undefined Stable Semantics for Probabilistic Logic Programs and Probabilistic Argumentation." In: *International Conference on Principles of Knowledge Representation and Reasoning (KR2022)*, 309–319.

V. H. N. Rocha and F. G. Cozman. 2022b. "Bipolar Argumentation Frameworks with Explicit Conclusions: Connecting Argumentation and Logic Programming." In: *20th International Workshop on Non-Monotonic Reasoning (NMR2022)*. Vol. 3197, 49–60.

V. H. N. Rocha, I. C. Silveira, P. Pirozelli, D. D. Mauá, and F. G. Cozman. 2023. "Assessing Good, Bad and Ugly Arguments Generated by ChatGPT: a New Dataset, its Methodology and Associated Tasks." In: *Progress in Artificial Intelligence*. Ed. by N. Moniz, Z. Vale, J. Cascalho, C. Silva, and R. Sebastião. Springer Nature Switzerland, Cham, 428–440. ISBN: 978-3-031-49008-8.

T. Sato. 1995. "A statistical learning method for logic programs with distribution semantics." In: *Conference on Logic Programming*, 715–729.

F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini. 2009. "The Graph Neural Network Model." *IEEE Transactions on Neural Networks*, 20, 1, 61–80.

E. Shnarch, L. Choshen, G. Moshkowich, R. Aharonov, and N. Slonim. Nov. 2020. "Unsupervised Expressive Rules Provide Explainability and Assist Human Experts Grasping New Domains." In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. Ed. by T. Cohn, Y. He, and Y. Liu. Association for Computational Linguistics, Online, (Nov. 2020), 2678–2697. DOI: 10.18653/v1/2020.findings-emnlp.243.

J. Si, L. Sun, D. Zhou, J. Ren, and L. Li. Mar. 2023. "Biomedical Argument Mining Based on Sequential Multi-Task Learning." *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 20, 2, (Mar. 2023), 864–874. DOI: 10.1109/TCBB.2022.3173447.

W. Song, Z. Song, R. Fu, L. Liu, M. Cheng, and T. Liu. Nov. 2020. "Discourse Self-Attention for Discourse Element Identification in Argumentative Student Essays." In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Ed. by B. Webber, T. Cohn, Y. He, and Y. Liu. Association for Computational Linguistics, Online, (Nov. 2020), 2820–2830. DOI: 10.18653/v1/2020.emnlp-main.225.

C. Stab and I. Gurevych. Sept. 2017. "Parsing Argumentation Structures in Persuasive Essays." *Computational Linguistics*, 43, 3, (Sept. 2017), 619–659. DOI: 10.1162/COLI_a_00295.

M. Stahl, N. Michel, S. Kilsbach, J. Schmidtke, S. Rezat, and H. Wachsmuth. June 2024. "A School Student Essay Corpus for Analyzing Interactions of Argumentative Structure and Quality." In: *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. Ed. by K. Duh, H. Gomez, and S. Bethard. Association for Computational Linguistics, Mexico City, Mexico, (June 2024), 2661–2674. DOI: 10.18653/v1/2024.naacl-long.145.

P. Stapleton and Y. (Wu. 2015. "Assessing the quality of arguments in students' persuasive writing: A case study analyzing the relationship between surface structure and substance." *Journal of English for Academic Purposes*, 17, 12–23. DOI: https://doi.org/10.1016/j.jeap.2014.11.006.

A. Toledo, S. Gretz, E. Cohen-Karlik, R. Friedman, E. Venezian, D. Lahav, M. Jacovi, R. Aharonov, and N. Slonim. Nov. 2019. "Automatic Argument Quality Assessment - New Datasets and Methods." 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). Ed. by K. Inui, J. Jiang, V. Ng, and X. Wan. Association for Computational Linguistics, Hong Kong, China, (Nov. 2019), 5625–5635. DOI: 10.18653/v1/D19-1564.

P. Totis, A. Kimmig, and L. D. Raedt. 2021. *SMProbLog: Stable Model Semantics in ProbLog and its Applications in Argumentation*. Tech. rep. arXiv:2110.01990. arXiv.

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

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. 2017. "Attention Is All You Need." In: *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*, 6000–6010.

P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio. 2018. "Graph Attention Networks." In: *Proceedings of the 6th International Conference on Learning Representations (ICLR)*.

H. Wachsmuth, K. Al-Khatib, and B. Stein. Dec. 2016. "Using Argument Mining to Assess the Argumentation Quality of Essays." In: *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. Ed. by Y. Matsumoto and R. Prasad. The COLING 2016 Organizing Committee, Osaka, Japan, (Dec. 2016), 1680–1691.

H. Wachsmuth, G. Lapesa, E. Cabrio, A. Lauscher, J. Park, E. M. Vecchi, S. Villata, and T. Ziegenbein. May 2024. "Argument Quality Assessment in the Age of Instruction-Following Large Language Models." In: *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*. Ed. by N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti, and N. Xue. ELRA and ICCL, Torino, Italia, (May 2024), 1519–1538.

H. Wachsmuth, N. Naderi, Y. Hou, Y. Bilu, V. Prabhakaran, T. A. Thijm, G. Hirst, and B. Stein. Apr. 2017. "Computational Argumentation Quality Assessment in Natural Language." In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. Ed. by M. Lapata, P. Blunsom, and A. Koller. Association for Computational Linguistics, Valencia, Spain, (Apr. 2017), 176–187.

H. Wachsmuth, B. Stein, and Y. Ajjour. Apr. 2017. ""PageRank" for Argument Relevance." In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. Ed. by M. Lapata, P. Blunsom, and A. Koller. Association for Computational Linguistics, Valencia, Spain, (Apr. 2017), 1117–1127.

P. Walley. 1991. *Statistical Reasoning with Imprecise Probabilities*. Chapman and Hall, London.

H. Yannakoudakis and R. Cummins. June 2015. "Evaluating the performance of Automated Text Scoring systems." In: *Workshop on Innovative Use of NLP for Building Educational Applications*. Association for Computational Linguistics, Denver, Colorado, (June 2015), 213–223.

H. Zhang and D. Litman. Apr. 2021. "Essay Quality Signals as Weak Supervision for Source-based Essay Scoring." In: *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*. Ed. by J. Burstein, A. Horbach, E. Kochmar, R. Laarmann-Quante, C. Leacock, N. Madnani, I. Pilán, H. Yannakoudakis, and T. Zesch. Association for Computational Linguistics, Online, (Apr. 2021), 85–96.

A Background Concepts

The main text assumes familiarity with several technical topics. For completeness, this appendix provides concise overviews of the core concepts used throughout the paper and points to key references for further study.

A.1 Transformer Architecture

The Transformer architecture introduced by Vaswani et al. (2017) marked a major shift in neural sequence modeling by replacing recurrence and convolution with a multi-head self-attention mechanism. This mechanism builds on the attention concept first proposed by Bahdanau et al. (2015), which allows a model to dynamically focus on relevant parts of the input rather than compressing the entire input into a single fixed-length vector. By computing a context vector at each decoding step, attention mechanisms enable significantly better handling of long-range dependencies.

In a Transformer, self-attention allows each token in a sequence to attend to all other tokens in parallel, capturing global contextual relationships without the sequential bottleneck of RNNs or LSTMs. Multi-head attention extends this idea by learning multiple attention patterns simultaneously, enabling the model to represent different types of linguistic or semantic relationships.

Because attention alone is position-invariant, Transformers incorporate positional encodings to preserve token order. Stacking self-attention and feed-forward layers produces deep contextual representations of sequences.

The Transformer architecture underlies most contemporary pretrained language models, including:

- **Encoder-only models** (e.g., BERT (Devlin et al. 2019)) for representation learning.
- **Decoder-only models** (e.g., GPT (Brown et al. 2020)) for generative modeling and reasoning.

- **Encoder–decoder models** (e.g., T5 (Raffel et al. 2020)) for sequence-to-sequence tasks.

Due to its scalability, efficiency, and interpretability through attention patterns, the Transformer has become the dominant architecture across NLP, vision, and other settings.

A.1.1 Encoder Language Models. Encoder language models process an entire input sequence bidirectionally to produce contextualized token embeddings. The most influential example is BERT (Devlin et al. 2019), which introduced the masked language modeling pretraining objective. Encoder models are commonly used as backbone components for downstream tasks such as classification, natural language inference and argument mining.

A.1.2 Token Classifiers. Token classifiers assign labels to individual tokens (or subword units). Classical approaches combine recurrent networks with structured prediction layers such as CRFs (Lample et al. 2016). Modern architectures typically use encoder-based transformer representations followed by a classification layer. Tasks such as named-entity recognition and argumentative component segmentation are commonly addressed using this paradigm.

A.1.3 Large Language Models. Large Language Models (LLMs) are transformer-based language models scaled to billions of parameters and trained on extensive text corpora. Their increased capacity enables the emergence of strong generalization abilities, allowing them to perform many tasks with minimal or no task-specific supervision. A landmark example is GPT-3 (Brown et al. 2020), which demonstrated in-context learning capabilities at scale. LLMs are now widely used in natural language generation, reasoning, and interactive applications.

A.2 Graph Neural Networks

Graph Neural Networks (GNNs) operate over graph-structured data by iteratively aggregating information from a node's neighbors. The general framework was introduced by Scarselli et al. (2009). A widely adopted specialization is the Graph Convolutional Network (GCN), which propagates and transforms node features through neighborhood-based convolution-like operations (Kipf and Welling 2017).

GNNs are well-suited to argumentation because argument structures, including attacks and supports, form graph topologies.

A.2.1 Graph Attention Networks (GATs). Graph Attention Networks introduce attention-based message passing, allowing the model to learn the relative importance of each neighboring node (Veličković et al. 2018). Unlike GCNs, which weight all neighbors equally, GATs compute attention coefficients dynamically during training. This allows more expressive representations, particularly in heterogeneous or noisy argumentative structures.

B Details on the Argument Mining Module Development

During the development of our AM Module, several different ideas and attempts were made before we surpassed the state-of-art on the AAEC dataset. Below, we provide details on those approaches, including the ones that were not entirely successful (Approaches I, II and III), but that may provide insights to interested readers. In addition, we provide in B.7, a comparison between the attempts detailed before, leading to the decisions on the final AM Module.

B.1 Approach I: Token Classification Approach

Most AM methods rely on sequence classification, where sentences or sentence pairs are classified independently. This approach, however, ignores contextual dependencies, meaning the same sentence or relation can serve different functions – e.g., as a premise or claim or as an attack or support – depending on its role within the broader argument structure.

To address this limitation, the first AM approach for MIDIQE reframed all AM tasks as token classification problems, allowing the model to process all argument components or relations simultaneously while making use of the surrounding context. The tasks were redefined as:

(1) **Span Identification**: Each token is classified as either *B* (Begin), *I* (Inside) or *O* (Outside), allowing the reconstruction of argument components;
(2) **Component Classification**: Components are classified simultaneously within a text. The model receives all components of a text, separated by a designated token, and each token is labeled as *MC* (Major Claim), *P* (Premise), or *O* (Outside). A component is assigned a label based on the majority token classification;
(3) **Link Detection**: This task determines whether a component is linked to another one. The model receives a list with the source component, followed by all other components, including the source itself. Tokens are labeled as *L* (Link), *N* (No Link) or *O* (Out), with the majority label determining the presence of a link;
(4) **Relation Classification**: Given detected links, the model classifies relations as *A* (attack), *S* (support) or *O* (out), with the majority label defining the relation type. The modeling is similar to Link Detection, with links being classified simultaneously.

Each task was trained using HuggingFace's token classification framework,[11] with standard hyperparameters and a batch size of 8, a learning rate of 2×10^{-6}, and 20 training epochs.

B.2 Approach II: A Cross-Domain Argument Mining Dataset

In this second approach, the models from Approach I are trained again, but this time using a new combined dataset approach. This section explains the new dataset and its application.

AM datasets are often small and annotated in very different ways, making it difficult to develop robust AM models. While larger datasets have improved NLP tasks, acquiring additional AM annotations is expensive. Prior research suggests that diverse training data enhances performance across domains (Morio et al. 2022), yet previous approaches have overlooked the similarities between the arguments in all datasets.

To address this, we introduce a unified annotation methodology that standardizes argument structures across datasets. This enables the construction of a cross-domain "combined dataset" by translating existing annotations into a shared framework:

(1) Span Identification remained unchanged as existing frameworks reliably detect argumentative parts;
(2) Component Classification was simplified under the assumption that every argument defends one or a few main points. These were labeled as *MajorClaim*, while all other components were classified as *Premise*;
(3) Relation Classification annotations were inspired by Bipolar Argumentation Frameworks (Cayrol and Lagasquie-Schiex 2013), distinguishing only attack and support relations.

Existing datasets were adapted to fit this *minimalist framework*, as follows:

(1) **ArGPT** (Rocha, Silveira, et al. 2023): Already aligned with the minimalist framework, demanding no adaptations for integration into the combined dataset;
(2) **AAEC** (Stab and Gurevych 2017): The annotation framework of this dataset closely aligns with the minimalist framework, though *Claim* components were reclassified as *Premise*;
(3) **MTC** (Peldszus and Stede 2016): Preprocessing methods from the literature (Kuribayashi et al. 2020) were employed. The argument tree structure was used to classify the root component as *MajorClaim*, while all others were labeled *Premise*. The original relation annotations *Add*, *Example* and *Support* were merged into *Support*, while *Undercut* and *Rebut* were changed to *Attack*;

[11] https://huggingface.co/

(4) **CDCP** (Park and Cardie 2018): The original labels *Value* and *Policy* were mapped to *MajorClaim*, while *Fact*, *Testimony* and *Reference* became *Premise*. The original relations, *Reason* and *Evidence*, were reclassified as *Support*;
(5) **AbstRCT** (Mayer et al. 2020): The labels *Claim* and *Evidence* were relabeled as *Premise*. Meanwhile, the *Support* label was retained, and other relations were translated to *Attack*;
(6) **AASD** (Accuosto and Saggion 2020): Similar to MTC, the root component in the tree structures was classified as *MajorClaim*, with all others as *Premise*. For the relations, all labels except *Attack* (which was retained) were classified as *Support*;
(7) **ASRD-Debater** (Shnarch et al. 2020): Only sentences containing arguments were considered, reducing the number of examples from 700 to 260. This dataset was applicable only to the span identification task;
(8) **CaE-Debater** (Rinott et al. 2015): Similar to MTC and AASD, the root components were labeled as *MajorClaim* and other nodes as *Premise*. The original relations between claims and evidence were retained for link detection. After removing duplicates and overlaps, approximately 43% of the dataset was used;
(9) **Araucaria** (Reed 2006): Only fully annotated English samples were used (approximately 67% of the dataset). Nodes with annotated relations or combined arguments were eliminated, and the edges were translated accordingly. For example, two consecutive attacks became a single support after removing the middle node. The data was successfully adapted for all AM tasks, using the tree structure to classify components and the provided annotations to determine relations.

Following these adaptations, the final combined dataset consisted of 2852 texts, 23859 components, and 16435 relations across diverse text types (essays, abstracts, user comments, etc) and structures (tree and graph).

In this second approach, the models from Approach I - based on Token Classification - were first pretrained for 10 epochs on the combined dataset, then fine-tuned for 10 epochs on the target dataset (AAEC), using the same hyperparameters as before.

B.3 Approach III: Multi-task AM Model Details

This appendix outlines the methodology used to develop the Multi-Task Argument Mining Model (MTAM) Combined model, another approach we explored for the AM Module. Four key factors were investigated: (1) Encoder selection, (2) Hyperparameter tuning, (3) Number of epochs, and (3) Training strategy optimization. The approach for each of these factors is detailed next, with all models and tests implemented using the HuggingFace Framework.

Previous research has explored multi-domain and multi-task approaches for AM (Huang et al. 2023; Morio et al. 2022), but these methods have key limitations: they rigidly couple tasks, overlook dataset similarities, and rely on limited contextual information. Building on previous Approaches I and II, this approach introduced a modular multi-task model, trained on the combined multi-domain dataset, to address these issues.

Our MTAM balances task-specific modularity with multi-task learning. Its architecture consists of: (1) A shared encoder (DeBERTaV3-Large) that learns a general argumentation framework; and (2) Task-specific heads for Span Identification, Component Classification, Link Detection, and Relation Classification, enabling simultaneous training while allowing single-task operation if needed.

The MTAM development process involved selecting DeBERTaV3-Large as the encoder, choosing the number of training epochs - with 40 epochs selected for pre-training, and testing curriculum learning strategies to optimize performance.

B.3.1 Choosing the Encoder type. The following transformer-based encoders were evaluated: RoBERTa (Base and Large) were selected as baseline options due to their simplicity. Longformer (Base and Large) were included to evaluate whether handling longer texts would improve performance. Additionally, DeBERTaV3-Large was chosen to assess the impact of using a more recent encoder.

Table 10. Encoder type comparisons for the MTAM-Model.

	Span	Component		Link	Relation			Time
	F1	F1	Macro	F1	F1	Macro	Epochs	Hours
RoBERTa-Base	66.60	89.24	79.65	36.31	90.57	75.75	20	**5.45**
RoBERTa-Large	72.25	90.44	81.34	41.81	91.20	77.54	20	10.54
RoBERTa-Large	**79.95**	91.76	84.62	44.71	92.15	79.95	70	45.30
Longformer-Base	65.63	89.13	79.46	33.70	90.32	73.47	20	63.94
Longformer-Large	69.49	89.75	80.37	39.70	91.82	78.90	20	60.50
DeBERTa-Large	76.05	**91.92**	**84.71**	**46.84**	**93.83**	**84.39**	20	42.90

The models were trained on the cross-domain dataset using the minimalist framework for annotation. Table 10 presents the performance metrics and training time for each encoder.

DeBERTaV3-Large outperformed all other models across all tasks trained for 20 epochs. While RoBERTa-Large exhibited competitive performance and required less training time (10.5 hours vs. 42.9 hours), further experiments showed that even with extended training (70 epochs), RoBERTa-Large could not match DeBERTaV3's overall effectiveness. Consequently, DeBERTaV3-Large was chosen as the encoder for further development.

B.3.2 Hyperparameter Selection. Early experimentation indicated that multi-task learning models required more training epochs than single-task models to achieve optimal performance. The following hyperparameter settings were used across all experiments: learning rate fixed at 2×10^{-6}, a batch size of 8 and a dropout rate of 10% for the classification layers. Additionally, the loss function was rewritten (shown below), but equal weights for all tasks yielded the best results:

$$\mathcal{W}_{task} = \text{Number of Link Examples/Number of Task Examples.}$$

B.3.3 Determining the number of epochs. As multi-task learning involves simultaneous training on multiple tasks, the sequence and method of training can impact model performance. To explore this, eleven curriculum learning strategies were designed. Results indicated that some tasks benefited from joint training, while others performed better when trained separately. The optimal training strategy was found to be dependent on both the number of epochs and the selected encoder. Thus first the optimal number of training epochs was determined.

To identify that optimal number of training epochs, three evaluation metrics - that combined the F1-score from Span Identification and Link Detection with the Macro scores from other tasks - were considered: (1) Harmonic Mean: This metric balances performance across multiple tasks, inspired by the F1-score; (2) Weighted Mean: Weights were assigned to tasks based on dependencies (e.g., Span Identification, as a prerequisite for other tasks, was assigned higher importance); (3) Weighted Harmonic Mean: Combines the first two metrics to provide a balanced evaluation.

Figure 9 shows both the loss function (left axis, blue line) over the number of training epochs and the weighted harmonic mean for each epoch count (right axis, red line). The loss function decreases steadily but begins to plateau after approximately 40 epochs. Simultaneously, the weighted harmonic mean stabilizes around this point. Given the diminishing returns in performance beyond 40 epochs, this number was then determined to be the optimal number of training epochs.

B.3.4 The Curriculum Learning Strategies. Since the MTAM approach involves training multiple tasks using the same encoder as a foundation, the sequence in which these tasks are trained, their grouping during training, and various task combinations can influence final results. This section explores various strategies while maintaining the constraint that each task's dataset is used for 40 epochs, consistent with previous experiments.

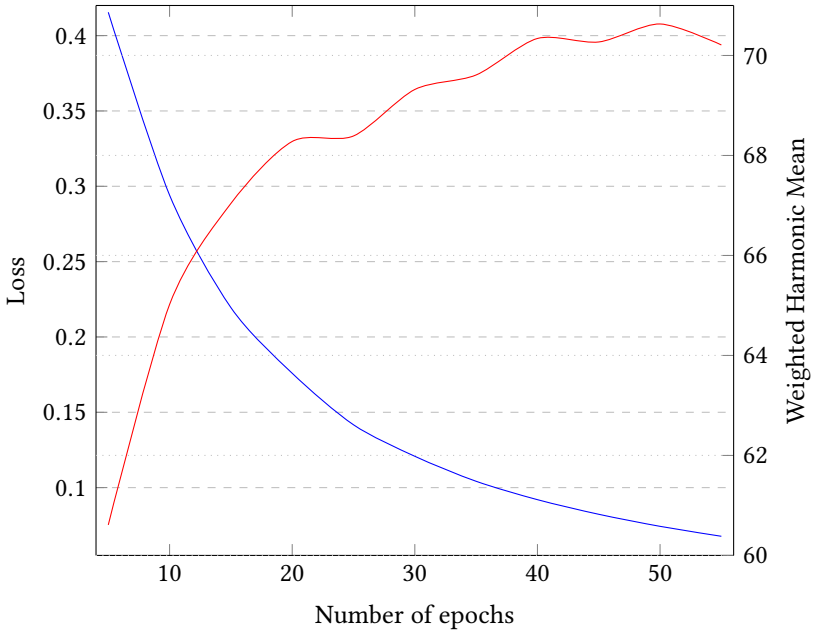

Fig. 9. (Left Axis) The loss function did decrease by increasing the number of epochs of training; (Right Axis) The increase in the number of training epochs did increase the performance of the model, but around 40 epochs, the results begin to oscillate around the same value.

Given the vast number of possible training configurations, it is impossible to explore every option. Instead, a set of strategies was designed, tested, and analyzed. The first issue was that some tasks are more challenging for the model to learn, yet training them together may still be beneficial. Based on this premise, the first three strategies were devised. Building on the results from the first these three strategies, three additional strategies were tested that grouped two AM tasks during training. Finally, the final five curriculum learning strategies integrated the insights from previous tests with new training approaches. All of those eleven strategies are summarized below:

(1) **Strategy I:** This strategy hypothesizes that the model will more effectively learn all argument mining tasks if trained in sequential order. The training sequence and the number of epochs per task are: Span (20), Component (20), Link (20), Relation (20), and then all tasks together (20);
(2) **Strategy II:** Initial tests indicated that the Component Classification task was the easiest to learn. Based on task difficulty rankings from early evaluations, the new order is: Component (20), Relation (20), Link (20), Span (20), and then all tasks together (20);
(3) **Strategy III:** Another set of early results suggested a different difficulty ranking. Thus, this strategy follows that different order: Component (20), Relation (20), Span (20), Link (20), followed by all tasks together (20);
(4) **Strategy IV:** Expanding on Strategy II, tasks were first trained separately, followed by the sequential combination of pipeline-dependent tasks: Component (10), Relation (10), Link (10), Span (10), Span/Component (10), Link/Relation (10), and All (20);
(5) **Strategy V:** Also inspired by Strategy II, this approach grouped tasks based on difficulty: Component and Relation were trained together as the easier tasks, while Link and Span were paired. The sequence was: Component (10), Relation (10), Link (10), Span (10), Relation/Component (10), Link/Span (10), and All (20);

Table 11. Multi Task Curriculum learning strategies.

Strategy	Span	Component		Link	Relation		Metrics		
	F1	F1	Macro	F1	F1	Macro	Harm.	Wght.	W.H.
I	79.29	92.76	86.46	47.54	93.27	82.95	69.85	73.76	69.62
II	78.94	92.54	86.21	49.31	94.37	85.62	**71.14**	74.46	**70.72**
III	79.23	92.40	86.10	47.90	94.15	85.19	70.36	74.11	70.00
IV	78.79	92.48	86.06	48.74	94.41	85.71	70.80	74.25	70.38
V	78.55	92.57	86.35	48.61	94.78	86.51	70.87	74.34	70.38
VI	79.53	92.75	86.64	48.15	94.27	85.48	70.69	74.45	70.33
VII	79.56	92.26	85.68	47.03	93.83	84.62	69.78	73.80	69.48
VIII	79.26	92.77	86.62	47.73	94.16	84.64	70.26	74.11	69.94
IX	79.72	92.69	86.51	47.94	93.64	84.12	70.36	74.21	70.09
X	80.15	92.76	86.66	48.08	94.03	84.73	70.65	**74.52**	70.37
XI	79.29	92.78	86.62	48.55	94.03	85.02	70.78	74.39	70.43
None	79.31	92.75	86.63	48.35	92.21	85.18	70.70	74.38	70.34

(6) **Strategy VI:** Given Strategy III's strong results, this approach prioritized Span Detection as a key task in the AM pipeline. It was sequentially combined with the other tasks before final joint training. The sequence was: Component (8), Relation (8), Span (8), Link (8), Span/Component (8), Span/Link (8), Span/Relation (8), Component/Link/Relation (16), and all tasks together (8);
(7) **Strategy VII:** Early results suggested that training Span Detection separately while grouping the other three tasks improved performance. The strategy followed: Span (20), Component/Link/Relation (20), All (20);
(8) **Strategy VIII:** Designed to mirror the AM pipeline, this approach progressively introduced tasks in their natural processing order: Span (10), Span/Component (10), Span/Link (10), Component/Link (10), Link/Relation (10), Component/Relation (10), Relation (10), All (10);
(9) **Strategy IX:** Building on Strategy VI, this approach omitted initial separate task training: Span/Component (10), Span/Link (10), Span/Relation (10), Component/ Link/Relation (20), All (10);
(10) **Strategy X:** This strategy integrated aspects of Strategies VI and VII to optimize performance: Span (10), Component/Link (10), Span/Relation (10), Component/Link/Relation (10), All (20);
(11) **Strategy XI:** Similar to Strategy X, this approach sought to blend key elements of successful strategies: Component (10), Span (10), Link (10), Span/Relation (10), Component/Link/Relation (10), All (20).

Table 11 presents the results of all strategies compared to a baseline (None), where all tasks are trained simultaneously. Overall, while all strategies, including the baseline, achieved similar results, some differences can be observed. For Span Identification, Strategy X achieved the best performance and also ranked among the top for Component Classification, followed by Strategies VI, VIII, and XI. For Link Detection, Strategy II outperformed all others by a significant margin, whereas Strategy V achieved the best results for Relation Classification. Taking into account the combined evaluation metrics, Strategy II performed best in two of them, while Strategy X led in one. Based on these findings, Strategy II was selected as the pretraining strategy for MTAM.

In conclusion, Strategy II emerged as the most effective curriculum learning strategy, balancing performance across all argument mining tasks. However, other strategies demonstrated strengths in specific tasks, suggesting that alternative approaches could be viable depending on the desired emphasis. With the encoder type, number of epochs, and training strategy selected, the MTAM approach was ready for final evaluation and comparison to other strategies.

B.4 Approach IV: Graph Neural Networks

Given that AM systems ultimately construct argumentation graphs, GNN-based approaches were investigated for all tasks except Span Identification, which solely relies on textual input. The models were built using the Deep Graph Library (DGL),[12] incorporating insights from existing Graph Attention Networks[13] (GAT) (Veličković et al. 2018) and training methodologies for node and edge classification.[14]

Each argumentative component's text was encoded using RoBERTa-Large embeddings, which were then used as node features in the GNN. The model architectures were adapted as follows:

(1) **Component Classification**: Built upon node classification techniques, this model replaced standard linear layers with LSTM layers to process tokenized text and applied the GAT architecture to better suit the problem. The classification layer integrates both graph embeddings and tokenized component representations for more accurate classification;
(2) **Link Detection and Relation Classification**: Modeled as edge classification tasks, these models use GAT architectures with LSTM-based layers for text processing. For Relation Classification, an additional linear layer combines information from both tokenized components and their graph embeddings. The other difference between the tasks lies in the labels used for classification.

Most default DGL parameters were retained, with modifications to hidden layers, learning rates (LR), and training epochs. Learning rates were optimized through a sweep between 0.2 and 2×10^{-6} over 1000 epochs. Stable models with decreasing training loss were prioritized, and among these, the best-performing were selected. Using the chosen LR, the number of epochs was optimized to ensure model convergence. The size of the hidden layers was also adjusted, with explorations ranging from 2 layers to 15 times the maximum size of the tokenized input. The optimal hidden layer size for each task was determined based on training time and performance improvements. Final training settings were:

- Component Classification: 500 epochs, LR = 0.2, 2 hidden layers;
- Link Detection: 300 epochs, LR = 2×10^{-3}, hidden layer size = 90;
- Relation Classification: 500 epochs, LR = 2×10^{-3}, hidden layer size = 0.5 × tokenized input size.

B.5 Approach V: Full-Text Context

Given the underwhelming performance of previous approaches (as detailed in experiments described in a later section), we revisited Approach I's rationale: a component or relation may receive different labels depending on context. However, instead of using token classification, Approach V adopted a simpler yet fundamentally different approach, incorporating full-text context for Component Classification, Link Detection, and Relation Classification (Span Identification remained unchanged, as it already processes the full text). For instance, in Component Classification, both the component and full-text context are provided as inputs. Similarly, Link Detection and Relation Classification incorporate the full text along with the pair of components.

For implementation, a RoBERTa-Large encoder was employed using HuggingFace's Sequence Classification framework. Models were trained with a learning rate of 2×10^{-6}, a batch size of 4, and 20 training epochs. Given these settings met performance goals, hyperparameter optimization is left for future work.

B.6 Approach VI: Cross-Domain Span Identification Details

As with Approach V, this sixth and last approach explored ideas in previous approaches in a different way. Specifically, this approach focused on span identification and, unlike previous attempts, it used the combined

[12] https://docs.dgl.ai/en/latest/index.html
[13] https://docs.dgl.ai/en/0.8.x/tutorials/models/1_gnn/9_gat.html
[14] https://docs.dgl.ai/en/latest/guide/training-node.html

Table 12. Tests to create the Span Detection Module. PT: Pretraining on combined dataset; FT: Fine-tuning on AAEC.

	Epochs PT	Epochs FT	F1
AAEC	0	20	75.71
AAEC + Combined	20	0	78.49
AAEC + Combined	50	0	83.71
AAEC + Combined	100	0	**85.42**
AAEC + Combined	25	75	84.14
AAEC + Combined	50	50	85.36
AAEC + Combined	75	25	85.38

Table 13. Comparison of the performance of all the approaches tested on this paper. Results are an average of several seeds.

	Span	Component		Link	Relation	
	F1	F1	Macro	F1	F1	Macro
Approach I - Token Clas.	70.17	74.49	67.38	29.29	90.30	70.17
Approach II - Combined data	72.52	75.01	67.58	32.34	90.83	71.27
Approach III - MTAM	**84.68**	76.48	71.06	39.77	94.16	83.17
Approach IV - GNN	-	77.99	73.90	**70.89**	71.37	49.07
Approach V - Full-Text	-	**86.41**	**84.95**	**70.35**	**96.77**	**90.62**
Approach VI - Cross-Domain Span	**85.42**	-	-	-	-	-

data set from Approach II, but changed the proportion of data from pre-training and fine-tuning. Furthermore, this approach systematically examined the impact of different total numbers of training epochs.

Table 12 presents results on the AAEC test set, with values averaged over five random seeds. The HuggingFace default hyperparameters were used, with a learning rate of 2×10^{-6} and a batch size of 4.The first two rows indicate that pretraining with the combined dataset improves performance compared to training solely on AAEC. The next three rows show that increasing the number of pretraining epochs continues to enhance performance, stabilizing around 100 epochs. Finally, the last four rows explore different pretraining-to-fine-tuning ratios, leading to the final Span Identification model: trained for 100 epochs on the combined dataset without additional fine-tuning.

B.7 Defining the AM Module

Table 13 presents the performance of each approach in the AM tasks they were trained for, highlighting the best results achieved by each approach. While additional approaches could have been explored, the results indicate that the six adopted paths were sufficient to achieve the goal of developing an AM Module at least on par with, if not surpassing, the state of the art. Thus, no further ideas were pursued.

A comparison of the first three approaches, all of which applied Token Classification modeling, reveals that pretraining on the combined dataset improved performance across tasks and that Multi-task training and further boosted results. However, all three approaches performed significantly worse in Link Detection compared to the other approaches, despite achieving reasonable results in other AM tasks. This consistent underperformance in Link Detection suggests that the Token Classification approach – which relied on the model's ability to identify all links in a text simultaneously – was ineffective for this specific task. In contrast, MTAM delivered one of the best performances for Span Identification (slightly lower than Approach VI) and ranked second-best in Relation Classification.

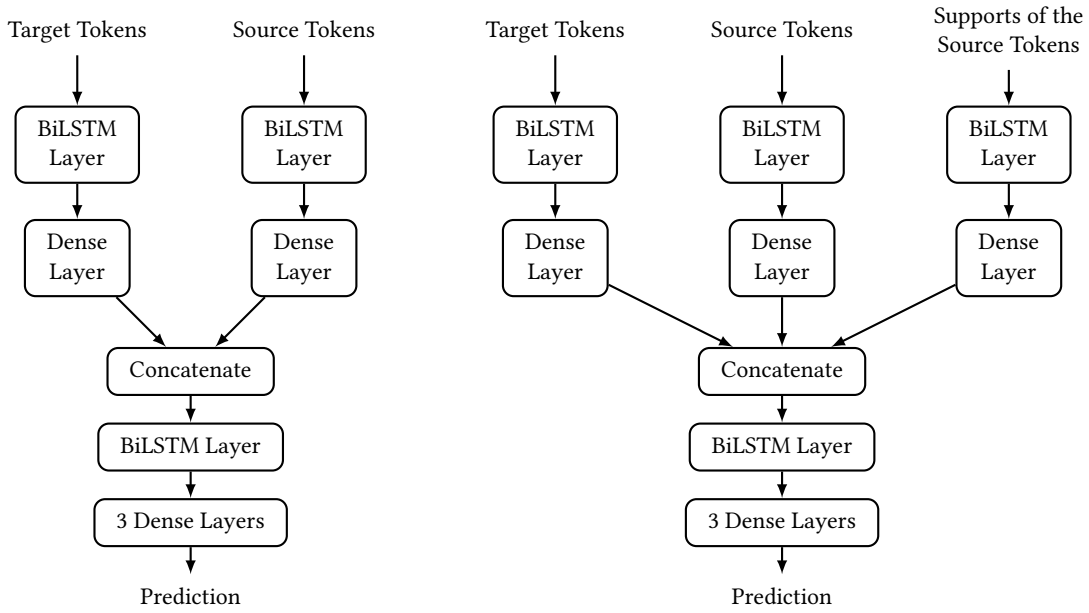

Fig. 10. Left: BiLSTM Model to predict the Cogency quality dimension; Right: BiLSTM Model to predict the Reasonableness quality dimension.

Results with Approach V indicate that providing full-text context is highly beneficial for Component and Relation Classification. Specifically, for Component Classification Approach V surpassed the GNN model (second-best approach) by nearly 11%, while for Relation Classification, it outperformed MTAM by approximately 7%. These improvements can be intuitively explained by the influence of context on both components and their relations. Thus, incorporating full-text context substantially enhances classification accuracy.

For Link Detection, the GNN-based approach (Approach IV) performed slightly better than the contextual model (Approach V), though the difference was only 0.5%. This may be due to the graph representation of argumentation being a constructed structure rather than an inherent textual feature, meaning explicitly modeling it as a graph yields better results.

To finalize the AM Module for the MIDIQE system, the best-performing model for each task was selected. Thus: (1) Span Identification: Model from Approach VI; (2) Component Classification: Full-Text Context model from Approach V; (3) Link Detection: GNN-based submodule from Approach IV; (4) Relation Classification: Contextual model from Approach V. This combination ensured state-of-the-art performance across all AM tasks, using the strengths of different modeling strategies.

C Single Argument Quality BiLSTM Model

As stated in the main text, two distinct architectures were tested for the SAQ module; however, only the Random Forest approach was successful. This section, however, provides more detail on the failed BiLSTM approach. Figure 10 illustrates the BiLSTM architectures for the Cogency and Reasonableness metrics.

Its implementation was made using the Keras framework[15] with default hyperparameters, training with a batch size of 32 for 30 epochs. These BiLSTM architectures were designed so that the model first processes the information from each group of components (target tokens, source tokens, and support tokens) separately before combining it into a unified classification network. This approach is inspired by the way humans might approach annotating such metrics.

D Argumentation Reasoning: Expanded Theoretical Foundations

This section of the appendix expands the theoretical foundations for the AR Module presented in the main text. It does not replace the reading of prior work (Rocha and Cozman 2022a,b), but it is intended to provide additional details for the readers.

As explained in the main text, the theoretical foundations of the AR Module are based on the Bipolar Conclusion-Augmented Argumentation Frameworks (Rocha and Cozman 2022b) and the L-credal semantics (Rocha and Cozman 2022a). These approaches were chosen for two key reasons: first, BCAFs establish a one-to-one equivalence between argumentation frameworks and logic programming, allowing the latter algorithms to be used for reasoning; second, the L-credal semantics is well-suited for probabilistic argumentation, as it ensures that the minimal number of conclusions remain undecided while maintaining an unbiased probability distribution. This subsection provides context for these theoretical foundations, beginning with an overview of argumentation frameworks to introduce BCAFs, followed by a discussion on probabilistic reasoning to justify the use of L-credal semantics. Most of the discussion in this subsection is based on the cited works (Rocha and Cozman 2022a,b) and additional information can be found there.

D.1 Argumentation Frameworks

D.1.1 Abstract Argumentation Frameworks. Abstract Argumentation focuses on how arguments and their relationships lead to conclusions, with no concern for the internal structure of arguments. Initially proposed by Dung (1995), an Abstract Argumentation Framework (AAF) consists of a set of arguments and a binary attack relation. This attack relation, written $A \rightarrow B$, implies that if A is accepted, B must be rejected. In the original proposal, the acceptability of arguments is determined using extension-based semantics; we use the equivalent *labeling semantics* (Baroni et al. 2004):

Definition 3. *A labeling $\mathcal{L}$ of an AAF is a function $\mathcal{L} : \mathcal{A} \rightarrow$ In, Out, Undecided, where arguments labeled* In *are accepted,* Out *are rejected, and* Undecided *are neither.*

A labelling is conflict-free iff there are no arguments A and B in the set of arguments labeled In *for which $A \rightarrow B$; and it is also admissible, iff for every argument A labeled* Out*, there exists an argument B labeled* In *such that $B \rightarrow A$.*

We then have a number of semantics:

Definition 4.

- *Complete semantics: An admissible argumentation labeling is complete iff: (i) if argument A is* accepted *then for all arguments B that attack A, B must be* rejected*; (ii) if argument A is* undecided *then there is no argument B that is* accepted *and attacks A, and there is an argument C that attacks A and is not* rejected*. A complete labeling is:*
  - *Preferred: iff the set of arguments labeled* In *is maximal;*
  - *Stable: iff the set of arguments labeled* Undecided *is empty;*
  - *Grounded iff the set of arguments labeled* In *is minimal;*

[15] https://keras.io/

– *Semi-stable iff the set of arguments labeled* Undecided *is minimal.*

Abstract argumentation frameworks as proposed by Dung (1995), based solely on attacks, do not fully capture human reasoning, as noted by Polberg and Hunter (2018). To address this, Bipolar Argumentation Frameworks (BAFs) were introduced by Cayrol and Lagasquie-Schiex (2013) to include support relations alongside attacks, offering a more nuanced view of argumentation. These support relations, written as $A \Rightarrow B$, can be interpreted in different ways, but here the focus in on the necessary interpretation: If B is accepted, A must also be accepted as well.

In BAFs, semantics incorporate indirect defeat, where an argument can be defeated indirectly through a sequence of supported arguments. Using this concept, all the relevant semantics can also be applied to BAFs.

D.1.2 Bipolar Conclusion-Augmented Argumentation Frameworks. The relationship between argumentation frameworks and other non-monotonic reasoning paradigms, especially logic programming, has been widely explored since Dung (1995). Logic programs provide a way to formalize reasoning processes through a set of rules, where each rule has a head and a body. The following in an example of a rule, where H, each A_i, and each B_i are atoms and the left-hand side H is the *head*, while the right-hand side is the *body*:

$$H \text{ :- } A_1, \ldots, A_m, \textbf{not } B_1, \ldots, \textbf{not } B_n.,$$

For a logic program $\mathbf{P}$, a three-valued interpretation $(\mathcal{T}, \mathcal{F})$ assigns each atom either to $\mathcal{T}$ (it is then true), to $\mathcal{F}$ (it is then false), or to neither of them (it is then undefined). The reduct of a logic program with respect to a three-valued interpretation $\mathcal{I}$, denoted $\mathbf{P}/\mathcal{I}$, simplifies the program by removing rules with negated atoms that are assigned true and adjusting for undefined atoms. A *three-valued model* of $\mathbf{P}$ is an interpretation $(\mathcal{T}, \mathcal{F})$ such that: (1) H is in $\mathcal{T}$ if there is a rule whose head is H and where each A is in $\mathcal{T}$; (2) H is in $\mathcal{F}$ if every rule whose head is H is such that there is some A in $\mathcal{F}$.

Given the definitions above, $\mathbf{P}/\mathcal{I}$ has a unique three-valued model with minimal $\mathcal{T}$ and maximal $\mathcal{F}$. This model is denoted $\Phi_{\mathbf{P}}(\mathcal{I})$. Several semantics are then defined.

Definition 5.

- *Partial stable model: is an interpretation $\mathcal{I}$ such that $\Phi_{\mathbf{P}}(\mathcal{I}) = \mathcal{I}$;*
- *Regular model: is the partial stable model with maximal $\mathcal{T}$;*
- *Well-founded model: is the partial stable model with minimal $\mathcal{T}$;*
- *Stable model: is a partial stable model without* undefined *atoms;*
- *L-stable model: is the partial stable model with the minimal number of* undefined *atoms.*

Translations between logic programs and AAFs can be performed using the WCG algorithm (Caminada et al. 2015). This algorithm processes rules to generate arguments, enabling the transformation of a logic program into an AAF. Once in this form, various argumentation semantics can be applied to assess argument acceptability, facilitating comparisons between argumentation frameworks and logic programming models. To enable such comparisons, it is necessary to map argument labels to the atom level (or "conclusion" level). This is done by assigning each conclusion the highest value among its associated arguments, following the order In > Undecided > Out (Caminada et al. 2015). Intuitively, this mapping ensures that each conclusion is represented by the argument that provides its strongest defense.

We have:

Theorem 1. *(Theorems 19 to 22 of (Caminada et al. 2015)). The labels of conclusions obtained by the semantics P-stable, regular, well-founded and stable in a logic program are equivalent to those obtained by the complete, preferred, grounded and stable semantics in an AFF respectively, with the subsequent transformation to the conclusions level.*

Logic programming typically relies either on the well-founded or on the stable semantics. The well-founded semantics ensures the existence of a unique model but fails to distinguish between situations where an atom remains unresolved due to structural constraints and cases where a decision could be made but is simply deferred. In contrast, the stable model semantics eliminates undefined values but may yield multiple models or none. These limitations are highlighted in the following example:

Example 8. *(Rocha and Cozman 2022a) Take the set of rules:*

$$G_1 \text{ :- } \textbf{not } G_2., \qquad G_2 \text{ :- } \textbf{not } G_1., \qquad H \text{ :- } G_1., \qquad H \text{ :- } G_2..$$

All atoms are left undefined *by the well-founded semantics, even if it is perfectly possible to label, for example,* G_1 *as* false *and* G_2 *as* true *or the other way around without disrespecting any structural constraints of the logic program. In both scenarios, which are also the stable models,* H *would get the* true *label and not the* undefined *label left by the well-founded semantics.*

Also consider the following logic program:

$$H \text{ :- } \textbf{not } H.$$

Atom H *cannot be* true *nor* false.

The well-founded semantics uses the same undefined *label for both situations, while the stable semantics fails to offer a semantics for the last case.*

In an argumentative setting, failing to derive conclusions due to the absence of stable models is undesirable, and at the same time, it is desirable to have a minimal set of conclusions left undecided (Rocha and Cozman 2022a). To address this, the *L-stable semantics* offers a compromise between well-founded and stable semantics. The L-stable model of a logic program is a partial stable model that minimizes undefined values while preserving the constraints of the logic program (**Sacca97**). If a logic program has one or more stable models, they coincide with its L-stable models. However, if no stable models exist, the L-stable semantics still provides P-stable models with the fewest possible undefined atoms.

Given a selection of the L-stable semantics, the notable exception to Theorem 1, i.e. the lack of equivalence between the semi-stable and L-stable semantics, becomes important.

To establish a correspondence between semi-stable and L-stable semantics, Claim-augmented Argumentation Frameworks (CAFs) have been introduced. In CAFs, each argument is explicitly associated with a conclusion (Dvorák, Rapberger, et al. 2020a,b; Dvorák and Woltran 2019; Rapberger 2020), a move that allows for the maximization/minimization of labels to happen directly at the conclusion level.

Although the original CAF frameworks use the term "claim", here "conclusion" is adopted to maintain consistency with the broader literature on argumentation frameworks and logic programming (Caminada et al. 2015), without changing the framework's structure. Furthermore, CAFs have also been expanded to include the support relation, resulting in the Bipolar Conclusion-Augmented Argumentation Framework (BCAF) (Rocha and Cozman 2022b), which have been shown to achieve a one-to-one equivalence with logic programming (Rocha and Cozman 2022b). A BCAF is defined as follows:

Definition 6. *(Rocha and Cozman 2022b) A Bipolar Conclusion-augmented Argumentation Framework (BCAF) is a tuple* $(\mathcal{A}, \mathcal{S}, f, \mathcal{R}_-, \mathcal{R}_+)$*, with* $\mathcal{A}$ *a finite set of arguments,* $\mathcal{S}$ *a set of conclusions,* f *a function of arguments to conclusions,* $\mathcal{R}_- \subseteq \mathcal{A} \times \mathcal{A}$ *the attack relation and* $\mathcal{R}_+ \subseteq \mathcal{A} \times \mathcal{A}$ *the support relation.*

In a BCAF, an argument $Arg \in \mathcal{A}$ can be defined in some different ways, but it is usually assumed that it is understood as a single rule $H \text{ :- } A_1, \ldots, A_m, \textbf{not } B_1, \ldots, \textbf{not } B_n.$. This way an argument Arg has *conclusion* H, *rules* $\{H \text{ :- } A_1, \ldots, A_m, \textbf{not } B_1, \ldots, \textbf{not } B_n.\}$, *vulnerabilities* $\text{Vul}(A) = \{B_1, \ldots, B_n\}$ and *necessities* $\text{Nec}(A) = \{A_1, \ldots, A_m\}$.

An attack relations in BCAFs is understood as follows: one argument attacks another if its conclusion is among the other's vulnerabilities. The support relation is, in this work, based on the necessary support model by Alfano et al. (2020):

Definition 7. *(Rocha and Cozman 2022b) An argument A supports another argument B iff $f(A) = a$ is a necessity of B. Thus, for B to be accepted, at least one of the supports A_i with $f(A_i) = a$ must also be accepted. If all $f(A_i) = a$ are rejected, then B is also rejected.*

Using this foundation, the complete conclusion labeling can be redefined for BCAF:

Definition 8. *(Rocha and Cozman 2022b) Let $(\mathcal{A}, \mathcal{S}, f, \mathcal{R}_-, \mathcal{R}_+)$ be a BCAF and $\mathcal{L}$ a conclusion labeling of $\mathcal{S}$. Then $\mathcal{L}$ is a* complete conclusion labeling *iff:*

(i) if a conclusion a is rejected *then for every argument $A \in f^{-1}(a)$ there is an argument B that attacks A with a* accepted *conclusion $f(B)$ and/or there is a necessity c in $Nec(A)$ labeled* rejected *because all arguments C with $f(C) = c$ are* rejected*;*

(ii) if a conclusion a is accepted *then for some argument $A \in f^{-1}(a)$, each argument B that attacks A must have a* rejected *conclusion $f(B)$ and for each necessity c in $Nec(A)$ there is at least one* accepted *argument C with $f(C) = c$;*

(iii) if a conclusion a is undecided *then there is no argument $A \in f^{-1}(a)$ for which all arguments B that attack A have $f(B)$ as* rejected *and/or there is no A for which all of its necessities $c \in Nec(A)$ are associated with at least one* accepted *argument C with $f(C) = c$; and for some argument $A \in f^{-1}(a)$, there is no argument B that attacks A with* accepted *conclusion $f(B)$, and there is an argument C that attacks that same A and whose conclusion $f(C)$ is not* rejected*; similarly for some argument $A \in f^{-1}(a)$, there is no necessity $b \in Nec(A)$ for which at least one argument in $f^{-1}(b)$ is* accepted*, and there is a necessity $c \in Nec(A)$ whose label is not* rejected*.*

The other semantics immediately follow from the complete semantics. To explore the equivalences between BCAFs and logic programming, two key properties are defined:

Definition 9. *(Rocha and Cozman 2022b) Let $AF = (\mathcal{A}, \mathcal{S}, f, \mathcal{R}_-, \mathcal{R}_+)$ be a BCAF. AF is said to be redundant if there is at least a pair of arguments $A, B \in \mathcal{A}$ such that $f(A) = f(B)$, Vul(A) = Vul(B) and Nec(A) = Nec(B).*

Definition 10. *(Rocha and Cozman 2022b) Let $AF = (\mathcal{A}, \mathcal{S}, f, \mathcal{R}_-, \mathcal{R}_+)$ be a BCAF. AF is said to be well-formed if all the arguments $A \in \mathcal{A}$ that hold the same conclusion $f(A)$ attack and support the same arguments.*

Throughout this section, it is assumed all BCAFs are well-formed and non-redundant. Given this, the WCG algorithm must be adapted to fit BCAFs. This can be done as follows:

(1) Starting with a rule set, process each rule at a time;
(2) If a rule of the form H :– **not** $B_1, \ldots,$ **not** B_n. is found, then generate an argument A with *conclusion* H, *rules* $\{H$:– **not** $B_1, \ldots,$ **not** $B_n.\}$ and *vulnerabilities* Vul(A) $\{B_1, \ldots, B_n\}$;
(3) If a rule of the form H :– $A_1, \ldots, A_m,$ **not** $B_1, \ldots,$ **not** B_n. is found, then generate an argument A with *conclusion* H, *rules* $\{H$:– $A_1, \ldots, A_m,$ **not** $B_1, \ldots,$ **not** $B_n.\}$, *vulnerabilities* Vul(A) $\{B_1, \ldots, B_n\}$ and *necessities* Nec(A) $\{A_1, \ldots, A_m\}$;
(4) After processing all the rules, the relations between arguments are established. If an argument A has a conclusion that is present in the vulnerabilities of another argument B, then A attacks B. On the other hand, if the conclusion of A is present in the necessities of B, then A supports B. With this, and with each argument being linked to a conclusion, the bipolar conclusion-augmented argumentation graph is created.

Figure 11 illustrates a BCAF generated using this method. Theorem 2 ensures that this adaptation preserves the equivalence between P-stable semantics in logic programming and complete semantics in BCAFs. Besides that, given that the maximization/minimization of labels is now done at the conclusion level for both formalisms, the semi-stable and L-stable semantics are considered equivalent.

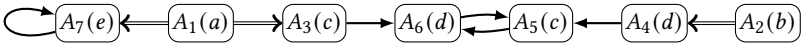

Fig. 11. A Bipolar Conclusion-augmented Argumentation Graph.

Theorem 2. *(Rocha and Cozman 2022b) Let* **P** *be a logic program and* $AF_P = (\mathcal{A}, \mathcal{S}, f, \mathcal{R}_-, \mathcal{R}_+)$ *the BCAF generated by the modified WCG algorithm. Then the complete, preferred, grounded, stable and semi-stable conclusion labels of* AF_P *are identical to the labels assigned respectively by the partial stable, regular, well-founded, stable and L-stable models of* **P**.

To translate a BCAF back into a logic program, the following procedure is applied:

Definition 11. *(Rocha and Cozman 2022b) Let* $\mathbf{C} = (\mathcal{A}, \mathcal{S}, f, \mathcal{R}_-, \mathcal{R}_+)$ *be a BCAF. For each argument A, a rule is generated as* $f(A) :- f(A_1), ..., f(A_m), \textbf{not } f(B_1), ..., \textbf{not } f(B_n)$. *where* $A_1...A_m$ *are the arguments that support A and* $B_1...B_n$ *are the ones that attack it.* $\mathbf{P}_\mathbf{C}$ *denotes the logical program constructed by this method.*

Both the forward and backward translations ensure that there is no loss of information, so the logic programs and non-redundant BCAFs generated in repeated translations are always the same. Hence the BCAFs achieve a one-to-one equivalence between argumentation (well-formed and non-redundant BCAFs) and logic programs. The results however are also relevant for redundant BCAFs, given that even though they may not map exactly to their original graphs after repeated translations, the underlying conclusion relationships remain unchanged, preserving semantic equivalence.

In short, BCAFs expand abstract argumentation by connecting each argument with its conclusion and adding the support relation. In doing that, it achieves a one-to-one equivalence between argumentation frameworks and logic programming, which allows one to use the semi-stable semantics for the first and L-stable semantics for the latter to produce equivalent results.

D.2 Probabilistic Argumentation

The computational argumentation models discussed so far operate under several key assumptions. They assume that individuals interpret arguments, claims, and conclusions in a uniform manner and that the relationships between components are clear and universally agreed upon. However, these assumptions do not always hold, as shown by an empirical study by Polberg and Hunter (2018). To address this, these authors propose that probabilistic argumentation should model individual opinions, such as in epistemic graphs (Hunter, Polberg, et al. 2020) and also structural uncertainties in argumentation graphs, such as in the constellation approach (Hunter and Noor 2020).

A proposal that fulfills that role and synthesizes elements from epistemic and constellation approaches is the SMProbLog approach (Totis et al. 2021). This framework models uncertainty by associating probabilities with selected facts, as is commonly done in probabilistic logic programming (Fierens et al. 2014; Sato 1995). While this approach does not capture all forms of subjective or objective uncertainty in argumentation, it strikes an effective balance between expressiveness and complexity – both in terms of computational demands and the cognitive load imposed on human users interacting with the formalism. The authors adopted ProbLog's syntax (Fierens et al. 2014) and specify a probabilistic fact using the syntax $\alpha :: F.$ to mean, intuitively, that with probability α the fact $F.$ is added to the program (and with probability $1 - \alpha$ the fact $F.$ is discarded). A set of rules and probabilistic facts is a *probabilistic logic program.*

In their work, Totis et al. (2021) proposed a semantics for probabilistic logic programming based on the stable semantics. This proposed semantics is directly based on Sato's distribution semantics (Fierens et al. 2014; Sato 1995). That means a *choice* for probabilistic fact $\alpha :: F.$, where F is a ground atom, is a decision as to whether the

probabilistic fact is replaced by the fact $F.$ or simply discarded. A *total choice* is a set of choices, one per probabilistic fact. All choices are assumed independent and for a set of probability facts $\{\alpha_i :: F_i.\}_{i=1}^{N}$, the probability of a total choice O is:

$$P(O) = \prod_{\substack{(\alpha_i :: F_i.) \\ \text{not discarded by } O}} \alpha_i \prod_{\substack{(\alpha_i :: F_i.) \\ \text{discarded by } O}} (1 - \alpha_i). \tag{1}$$

Because a probabilistic logic program is converted into a logic program by fixing a total choice, the probability distribution over total choices induces a probability distribution over logic programs. To ensure a comprehensive semantic foundation, the authors propose the adoption of stable semantics, which has been shown to be equivalent to its counterpart in argumentation frameworks (Caminada et al. 2015).

The reasoning procedure then consists of distributing the probability mass over the models generated by each total choice after applying the stable semantics. The probability of an atom is then defined as the sum of the probabilities of the models in which the atom is accepted. That might present a problem to traditional approaches to probabilistic logic programming, as one property of the stable semantics is that it can generate multiple stable models. To solve this, Totis et al. (2021) propose that the probability mass of a total choice should be distributed equally among its stable models and they justify this through the principle of maximum entropy.

The probabilistic argumentation framework introduced by Totis et al. (2021) successfully integrates features of earlier approaches by cleverly using the equivalences between logic programming and argumentation. This framework models several characteristics found in other argumentation frameworks, such as support relations and quantitative assessments of attacks and supports. By assigning probabilities to both atoms and rules, it fulfills the requirements laid out by Polberg and Hunter (2018). However its reliance on stable semantics poses challenges, since, as discussed earlier, the L-stable semantics produces more satisfactory outcomes in argumentation scenarios (Rocha and Cozman 2022a).

Building on the work of Totis et al. (2021), Rocha and Cozman (2022a) introduce the the credal least undefined stable (L-credal) semantics, which extends probabilistic logic programming to better suit argumentation frameworks.

As before, given a total choice, a probabilistic logic program is transformed into a logic program. Thus the probability distribution over total choices induces a probability distribution over logic programs. As we have mentioned, probabilistic logic programming typically relies on either the well-founded or stable semantics to interpret induced logic programs, but the L-stable semantics is better fitted to an argumentation setting.

Additionally, to handle scenarios with multiple stable (or least stable) models, previous works have proposed distributing probability mass uniformly over them (Baral et al. 2009; Totis et al. 2021). However, this "uniform probability" approach implicitly leads to preferences over beliefs and opinions (Walley 1991). Instead, Lukasiewicz (2005) suggested adopting a *credal semantics* that considers the full set of possible probability distributions over stable models (Augustin et al. 2014; Cozman and Mauá 2017). This credal approach ensures minimal commitment to specific probability assignments while maintaining flexibility in probabilistic argument evaluation. Combining the L-stable semantics with the credal semantics leads to the *credal L-stable semantics*, defined as follows:

Definition 12. *(Rocha and Cozman 2022a) Given a probabilistic logic program* **P***, its* credal least undefined stable *semantics, or* L-credal *semantics, is the set of all probability distributions over L-stable models of programs induced by total choices, such that each distribution agrees with the probabilities of total choices.*

The following example clarifies the semantics.

Example 9. *(Rocha and Cozman 2022a) Assume the following rules:*

$$c :\!- a.; \qquad d :\!- b., \qquad c :\!- \textbf{not } d., \qquad d :\!- \textbf{not } c.,$$

Table 14. Left: tight probability intervals for Example 9; if an interval contains a single number, the interval is replaced by the number. Right: probabilities using the SMProbLog approach by Totis et al. (2021). T = true, F = false, U = undefined and I = inconsistent.

	$v = T$	$v = F$	$v = U$
$\mathbb{P}(a = v)$	0.5	0.5	0.0
$\mathbb{P}(b = v)$	0.5	0.5	0.0
$\mathbb{P}(c = v)$	[0.5, 0.75]	[0.25, 0.5]	0.0
$\mathbb{P}(d = v)$	[0.5, 0.75]	[0.25, 0.5]	0.0
$\mathbb{P}(e = v)$	0.0	0.5	0.5

	$v = T$	$v = F$	$v = I$
$\mathbb{P}(a = v)$	0.0	0.5	0.5
$\mathbb{P}(b = v)$	0.25	0.25	0.5
$\mathbb{P}(c = v)$	0.125	0.375	0.5
$\mathbb{P}(d = v)$	0.375	0.125	0.5
$\mathbb{P}(e = v)$	0.0	0.5	0.5

$$e \text{ :- } a, \textbf{not } e., \qquad 0.5 :: a., \qquad 0.5 :: b..$$

There are four total choices ($\emptyset$, $\{a\}$, $\{b\}$, $\{a, b\}$), each with probability 0.25. *For total choice $\emptyset$, the induced logic program has two L-stable models (both are also stable models): one where every atom is* false *except c that is* true, *and another where every atom is* false *except d that is* true. *For total choice $\{a\}$, there is a single L-stable model (that is also the well-founded model): e is* undefined, *while both a and c are* true *and both b and d are* false. *For total choice $\{b\}$, there is again a single L-stable model (that is the single stable model): a and c and e are* false *and both b and d are* true. *Finally, for total choice $\{a, b\}$ there is a single L-stable model (that is the well-founded model): all atoms are* true *except e that is* undefined. *By adding probabilities across models for all possible distributions in the semantics, the tight probability intervals in Table 14 are obtained (left).* □

It is relevant to note that, differently from the well-founded and stable semantics, the L-credal semantics nicely differentiates between: (i) the uncertainty captured by the probabilities in probabilistic facts; (ii) the uncertainty regarding multiple stable models, captured by probability intervals; (iii) the uncertainty arising from inability to satisfy constraints, leading to non-zero probability for undefined values.

It is also interesting to compare the L-credal semantics results with those obtained by the SMProbLog approach. Table 14 (right) shows the probabilities obtained via the SMProbLog approach. Notably, the probabilities of inconsistent values are remarkably high, even for atoms that can be assigned other values.

In order to avoid controversies related to mixtures of probabilities and three-valued logic, we adopt the labels accepted rather than true, rejected rather than false and undecided rather than undefined. We note that the complexity of inferences using the proposed L-credal semantics has been determined to be $\mathsf{PP}^{\Sigma_2^P}$-complete (Rocha and Cozman 2022a).

E Baseline Prompts

To evaluate GPT-4.0 and Deepseek-R1 in Section 5, a trial-and-error process was used to develop effective prompts. The final prompt yielded the best observed performance for both models. We could not obtain better prompts despite trying a number of changes, even though one might always try to improve the prompts with additional trial-and-error effort.

The final prompt used for both models is as follows:

Example 10. *Are you familiar with the work "Computational argumentation quality assessment in natural language" by Henning Wachsmuth, Nona Naderi, Yufang Hou, Yonatan Bilu, Vinodkumar Prabhakaran, Tim Alberdingk Thijm, Graeme Hirst, and Benno Stein?*

In this work, the authors introduce a taxonomy of argumentation quality divided as:

- *Cogency: evaluates an argument's logical soundness. It operates more on the level of a single argument's quality rather than the quality of the whole of argumentation. For an argument to be cogent, it must have:*

  - *Local acceptability: The premises should be credible and rationally believable;*
  - *Local relevance: Each premise should contribute directly to accepting or rejecting the argument's conclusion;*
  - *Local sufficiency: The premises, when considered together, should provide enough support to make the conclusion rationally defensible.*
- *Reasonableness: assesses the argument's contribution to resolving the issue at hand. It operates in both the single argument level as well as the whole of argumentation. An argument(ation) is deemed reasonable if:*
  - *Global acceptability: The audience finds the argument(ation) acceptable as a fair and valid contribution to the issue.*
  - *Global relevance: The argument(ation) offers information that moves toward a resolution, helping guide the audience to an ultimate conclusion.*
  - *Global sufficiency: The argument(ation) anticipates and sufficiently rebuts potential counterarguments to support its stance.*
- *Effectiveness: examines the rhetorical appeal and clarity of the argument, including its ability to persuade and engage the audience, often influenced by emotional or stylistic elements. This dimension includes:*
  - *Credibility: The argument should present information in a way that builds trust in the author's reliability;*
  - *Emotional appeal: Persuasive arguments effectively evoke emotions, making the audience more receptive to the author's claims.*
  - *Clarity: Clear and straightforward language is crucial, avoiding unnecessary complexity or ambiguity.*
  - *Appropriateness: The style and tone of language should enhance credibility and emotional impact, while remaining suitable to the issue.*
  - *Arrangement: Properly arranged arguments introduce the issue, present supporting points and, in that order, their conclusion.*

The authors also specify that the Reasonableness dimension is dependent on the Cogency one, while Effectiveness depends on both.

In my PhD thesis, I am creating a new criterion called the Minimal Dialectical Criterion. This new dimension fits within the taxonomy previously discussed and it fits between Cogency and Reasonableness, i.e. it depends on the first while the latter depends on it.

The definition of the new dimension is: a good argumentation must at least be able to successfully defend/support its own main points (or major claims).

Given this, you will be given a text that should be classified as either good or bad, depending on its minimal dialectical quality. Thus, it should be labeled "good" if the text is able to clearly justify its main points (Major Claims) without contradictions and "bad" otherwise. The factual validity of the MajorClaims should not be taken into account, only if by itself the text is able to justify them.

The Major Claims, or main points, of the text are what the text is trying to convince the reader about. Remember where the criterion fits within the presented taxonomy.

Consider the following examples:

Example 1: "Should students be taught to compete or to cooperate? It is always said that competition can effectively promote the development of economy. In order to survive in the competition, companies continue to improve their products and service, and as a result, the whole society prospers. However, when we discuss the issue of competition or cooperation, what we are concerned about is not the whole society, but the development of an individual's whole life. From this point of view, I firmly believe that we should attach more importance to cooperation during primary education. First of all, through cooperation, children can learn about interpersonal skills which are significant in the future life of all students. What we acquired from team work is not only how to achieve the same goal with others but more importantly, how to get along with others. During the process of cooperation, children can learn about how to listen to opinions of others, how to communicate with others, how to think comprehensively, and even

how to compromise with other team members when conflicts occurred. All of these skills help them to get on well with other people and will benefit them for the whole life. On the other hand, the significance of competition is that how to become more excellence to gain the victory. Hence it is always said that competition makes the society more effective. However, when we consider about the question that how to win the game, we always find that we need the cooperation. The greater our goal is, the more competition we need. Take Olympic games which is a form of competition for instance, it is hard to imagine how an athlete could win the game without the training of his or her coach, and the help of other professional staffs such as the people who take care of his diet, and those who are in charge of the medical care. The winner is the athlete but the success belongs to the whole team. Therefore without the cooperation, there would be no victory of competition. Consequently, no matter from the view of individual development or the relationship between competition and cooperation we can receive the same conclusion that a more cooperative attitudes towards life is more profitable in one's success."

The Major Claims of the text are: (1) "we should attach more importance to cooperation during primary education" (2) "a more cooperative attitudes towards life is more profitable in one's success" - Reasoning: Since both of Major Claims are properly justified by the text, according to the Minimal Dialectical Criterion, it must be labeled good. - Answer: Good

Example 2: "Lessons with teachers versus others sources. Over the last half century the change in the life of human beings has increased beyond our wildest expectations. There are countless new technological equipments flooding into our lives, such as the Internet, television and cell phone. They all enrich our daily lives. Now, many students learn from Internet after class and they find it quite helpful. Many people find that students at schools and universities still learn much more from lessons with teachers. They argue that teachers teach students how to write, read and calculate since they are in kindergarten. And when they are in primary school, teachers teach them more further knowledge, like writing skills and using computers. It is teachers that help them with obstacles in schooling. On the other hand, students gain most of the knowledge through teachers and learn what is right and wrong with the assistance of teachers. Those who feel that students learn far more from other sources, such as the Internet and television, firmly believe that within this sources students learn lots of things which they can't learn in classes. They can only input some key words and google it, and then there are numberless articles and websites related to it. In this case, students learn things easily. Moreover, they contend that good television programs do teach students. For instance, Discovery Channel has many instructive episodes. Students have knowledge of others cultures, outer space etc. Personally, I think that students learn far more from their teachers than from other source. Because not only do teachers teach us knowledge but also the skills to tell right from wrong. Just imagine: if a student can't even make out whether the source is reasonable or not, then how can he get the right information to help him to learn?"

With Major Claim: (1) "students learn far more from their teachers than from other source" - Reasoning: This text shoud be labeled bad, because, according to the Minimal Dialectical Criterion, the text falis to justify its main point. The points made to support the Major Claim are of average strength and the text fails to counter the objetction made to the main point, i.e. fails to counter the claim "students learn far more from other sources, such as the Internet and television". Thus this text is bad. - Answer: Bad

Given this, could you label the following text as either "good" or "bad" according to the Minimal Dialectical Criterion. Please, make sure to provide the answer with the label only ("good" or "bad", nothing more) and do not forget to use both of "good" and "bad":

{ ESSAY TO BE CLASSIFIED }

Received 06 February 2026; accepted 17 June 2026