






Evaluating LLMs on Argument Mining Tasks in Brazilian Portuguese Debate Data

David Eduardo Pereira   [Systems and Computing - Federal University of Campina Grande | david.pereira@ccc.ufcg.edu.br]

Daniela Thuaslar Simão Gomes  [Language Department - Federal University of Campina Grande | daniela.thuaslar@estudante.ufcg.edu.br]

Claudio E. C. Campelo  [Systems and Computing - Federal University of Campina Grande | campelo@dsc.ufcg.edu.br]

 Systems and Computing, Federal University of Campina Grande, R. Aprígio Veloso, 882, Universitário, Campina Grande, PB, 58429-900, Brazil.

Received: 01 April 2025 • Accepted: 29 July 2025 • Published: 31 October 2025

Abstract This study investigates Argument Mining (AM) in Brazilian Portuguese data, focusing on audio transcriptions of semi-structured debates. It proposes an experimental setup to evaluate the effectiveness of Large Language Models (LLMs) in AM tasks. The research addresses key challenges in the field, such as the lack of universally accepted definitions, the absence of a cohesive theoretical framework for dataset standardization, the limited availability of annotated datasets, and underexplored evaluation methods for Artificial Intelligence (AI) models, particularly LLMs. Aiming to bridge these gaps, especially in the underrepresented Brazilian Portuguese context, the study employs multiple prompt engineering strategies, including Single-Prompt, 2-Prompts, and 4-Prompts. The 4-Prompts approach, which integrates few-shot and chain-of-thought (CoT) prompting, demonstrated the best overall performance. The evaluated LLMs include ChatGPT-3.5 Turbo, ChatGPT-4, Gemini, LLaMA 70B, and Sabiá 3. Results show that while LLMs can achieve up to 74% F1 score in basic argument detection, their performance significantly drops in more complex AM tasks that require nuanced interpretation, with a maximum F1 score of 43%. Comparisons with Portuguese-specialized models such as Sabiá 3 revealed similar or inferior performance compared to multilingual models. Surprisingly, LLaMA 70B emerged as the best-performing model across most AM tasks. These findings underscore the need for continued development of AM methodologies and highlight the importance of expanding Natural Language Processing (NLP) research to languages beyond English.

Keywords: Argument Mining, Debate, LLM

1 Introduction

Debates have been an essential part of human communication and decision-making since ancient Greece [Bentahar *et al.*, 2010], where philosophers such as Socrates, Plato, and Aristotle engaged in discussions that profoundly influenced Western thought. Over time, debates have become a fundamental aspect of our daily lives, offering numerous benefits, including enhancing communication skills, fostering critical thinking, and improving persuasive abilities. Hence, one of the most crucial aspects of debating is the capacity to construct and present well-structured arguments, which is essential for effective discourse.

In this sense, the process of argumentation permeates our daily interactions and is indispensable in various domains, including scientific research, essay writing, legal trials, and everyday discussions such as defending a point of view or discussing ideas. Given this, argumentation has academic and professional significance, as it promotes the pursuit of facts and evidence to support claims, enhancing logical reasoning, problem-solving skills, and the ability to construct well-founded perspectives. Therefore, it is an indispensable component of informed decision-making and coherent discourse.

Argument Mining (AM) is a subfield of Natural Language Processing (NLP) that focuses on the automatic identification and analysis of argument structures in textual data. Although AM is a relatively recent field, with research beginning around 2014 [Cabrio and Villata, 2018], it has been applied to a variety of textual sources, including essays [Nguyen, 2018; Sazid and Mercer, 2022], scientific papers [Accuosto *et al.*, 2021; Al Khatib *et al.*, 2021; Stylianou and Vlahavas, 2021], legal texts [Westermann *et al.*, 2022], and news articles [Lavee *et al.*, 2019]. Furthermore, AM extends beyond written texts including spoken discourse, such as political debates speeches [Mancini *et al.*, 2022; Mestre *et al.*, 2021], and podcasts [Pojoni *et al.*, 2023].

Research in AM and debates has traditionally focused on the ones with predefined formats, rounds and time constraints (structured debates). Particularly in political, legal contexts and online discussions on platforms such as X (formerly Twitter) and Reddit have also been widely explored [Sousa *et al.*, 2021; Habernal and Gurevych, 2016; Boltužić and Šnajder, 2016; Chakrabarty *et al.*, 2019], with consolidated datasets providing detailed argumentation annotations, especially in political discourse [Haddadan *et al.*, 2019; Lippi and Torroni, 2016]. Furthermore, datasets covering group discussions, often featuring experienced participants, have been developed,

usually following rigid structures with limited rounds and time-constrained speeches [Mirkin *et al.*, 2018].

In contrast, individual, semi-structured, and spoken (ISS) debates—such as those found in educational settings, public forums, and interviews, remain underexplored despite their rich argumentative content. The lack of datasets and research that thoroughly examine the complexity of these ISS debates has hindered progress in discourse analysis and the development of Artificial Intelligence (AI)-driven models tailored to this form of communication.

Furthermore, there is a notable scarcity of available data and evaluation research for AM tasks in Brazilian Portuguese. Nowadays, the existing research mainly focus on English language. In this sense, the gap becomes even more significant when considering AM in the context of debates for Brazilian Portuguese. To the best of the authors’ knowledge, no previous research has investigated AM in debates in Brazilian Portuguese. To bridge this critical research gap, this study introduces an evaluation of AI models for AM tasks, utilizing the DEBISS corpus, a newly annotated dataset of Brazilian Portuguese debates, focused on individual, semi-structured, and spoken interactions.

Another fundamental challenge in AM is the absence of a universal framework for analyzing argumentative texts. Argumentative structures vary according to context, cultural background, and rhetorical style [Chen *et al.*, 2022], leading to inconsistencies in the definition of arguments and its components such as claims and premises [Kotelnikov *et al.*, 2022]. This variability complicates efforts to establish standardized methodologies for detecting and interpreting arguments.

Moreover, it is crucial to note that beyond the absence of a universally accepted data framework for AM purposes, the use of Large Language Models (LLM) in AM is still in early stages. Only a limited number of studies have employed LLMs for AM tasks, particularly for argument analysis in debates. Hence, the main objective of this research is to investigate and evaluate the application of AM techniques, using LLMs, on the DEBISS corpus composed by audio transcriptions of debate sections in Brazilian Portuguese data with several AM tasks annotation.

Specifically, this study aims to identify the argumentative features present in debate discourse, such as the use of arguments and its components, for example: claims, types of reasoning, premises, and other elements extracted from the transcription of debate audio recordings. Furthermore, efforts will focus on implementing LLMs as tools for identifying the aforementioned argumentative features. The results demonstrated a reasonable capability of LLMs to identify arguments but not its components. Considering this, overall results demonstrated 68% of F1 score for the most basic AM task and around 31% for other more nuanced AM tasks. Demonstrating a poor ability for the tested LLMs in most of the AM tasks.

Taking this into account, understanding argumentative structures in spoken debates is crucial not only for advancing research in AM but also for practical applications in discourse analysis. For example, improving the understanding of how arguments are constructed, developed, and challenged in real-world discussions. Moreover, previous research has focused primarily on written argumentation, spoken debates present

unique challenges, such as spontaneous argument formulation, interruptions, and varied discourse strategies, which remain underexplored. By evaluating the performance of LLMs in AM tasks of transcribed debate data, this research aims to bridge the gap between theoretical argumentation studies and practical applications in NLP.

The ability to accurately identify argumentative components in debates facilitates discourse analysis and supports the development of more advanced AI-driven debate assistants. Therefore, this study’s findings highlight the current limitations of LLM in AM, offering insights that can guide the development of improved methods for AM in debates. These include better model selection, prompt engineering techniques, and more effective training and fine-tuning strategies—ultimately fostering more precise and effective argument analysis in spoken discourse.

The contribution of this research begins by advancing the field of AM, with a specific focus on the underexplored context of ISS debates. In addition, it provides an evaluation of LLMs, which are still in the early stages of development for AM-related tasks. This research also extends its contributions to languages beyond English, where most existing studies are concentrated, by evaluating AM performance in Brazilian Portuguese data. Addressing these aspects, the study contributes not only to the general development of AM but also to the advancement of AM methodologies tailored to debates in Brazilian Portuguese.

To address these challenges, this study establishes the following objectives:

- Investigate the performance of LLMs in detecting and classifying argumentative structures through comparative analysis and benchmark creation.
- Understand the impact of prompt engineering techniques on LLM performance in AM tasks, optimizing their ability to process argumentative structures.
- Determine whether a model specifically designed for Portuguese outperforms non-specialized models and also assess its effectiveness in identifying and classifying argumentative structures.

To better investigate and achieve the objectives listed above, the following Research Questions (RQ) are addressed in this study using the results obtained through the proposed methodology:

- **RQ1:** How effective are LLM-based approaches in AM tasks for ISS debate data?
- **RQ2:** Which LLM model achieves the highest performance in AM tasks?
- **RQ3:** Does a Portuguese-specialized model achieve better results in AM tasks compared to non-specialized models?
- **RQ4:** Which prompt engineering techniques produce the most effective results for AM tasks?

2 Related Work

This section provides an overview of studies related to this research. First, it discusses research on AM, followed by

its main applications in debates. Finally, it focuses on the application of AM using LLMs.

2.1 Argumentation Theory

Argumentation theory and schemes are attracting attention from those interested in argumentation and AI, highlighting the interdisciplinary nature of the research field [Reed and Norman, 2003]. Efforts to formalize and compile studies on argumentation schemes can be found in the literature, and these efforts are not new, but there remains a challenge due to a absence of consensus [Walton *et al.*, 2008]. Researchers have proposed creating a taxonomy for the available argument models. They summarized the models into three categories: rhetorical, dialogical, and monological [Bentahar *et al.*, 2010].

According to the research, **monological** models are defined as those that focus on the internal structure (micro) of an argument and the relationships within its elements, rather than the relationships between different arguments. Meanwhile, **dialogical** models analyze the exchanges between multiple parties or speakers, such as the interactions between two speakers (macro) [Bentahar *et al.*, 2010]. The monological model addresses the microstructure by focusing on the internal structure, while the dialogical model considers the macro structure, analyzing the context between different arguments and speakers [Habernal and Gurevych, 2017]. The final model classification in the proposed taxonomy are **rhetorical** models. The main idea behind rhetorical models is to consider the audience's perspective on the argumentative discourse, focusing on its rhetorical persuasiveness. These models aim to establish an evaluative view of the propositions made by arguments, rather than focusing on the micro or macro structures of arguments.

Researchers have compiled a compendium of about 60 different types of argument schemes [Walton *et al.*, 2008], which includes defined rules, examples, and references for each model. This research highlights the application of these models to AI, showcasing tools for representing arguments according to the proposed schemes. However, it is important to note that this compendium is not exhaustive; instead, it aims to convey the complexity of the subject and provide valuable contributions.

2.2 Argument Mining

The main purpose of AM is to transform unstructured text data into a structured format by identifying argument structures. This process provides a comprehensive understanding of the arguments' format, their roles within the text, and the ultimate objective of the argument. Although the resulting structures can vary significantly depending on the intended application of the data, the core principles of AM remain consistent across different contexts, which is the process of structuring text data with argumentative characteristics.

To better understand AM, it is important to examine how this NLP area has gained researchers' attention. Prior to AM, several related fields contributed to its early development, including opinion mining, controversy detection, citation mining, and argumentative zoning [Lawrence and Reed, 2020]. Opinion mining, also known as sentiment analysis, involves

the computational study of opinions, sentiments, and emotions in text. It is a crucial tool for analyzing user-generated content such as reviews, blog posts, and social media discussions to extract insights valuable for businesses and individuals. While sentiment analysis focuses on positive or negative sentiments, opinion mining includes broader perspectives, often incorporating argumentative structures to refine sentiment detection.

Furthermore, controversy detection extends opinion mining by identifying topics and texts that reflect conflicting viewpoints, offering insights into more nuanced argumentative texts that are likely to foster more complex arguments. Moreover, citation mining involves labeling citations in scientific texts with their rhetorical roles, linking citation function to argumentative relations such as support or conflict.

Additionally, argumentative zoning classifies sentences in scientific papers by their rhetorical and argumentative roles, such as presenting goals, results, comparisons, or critiques, offering insights that aid in understanding argument structures. While these fields differ in their goals and methodologies, they provide essential techniques that serve as a valuable foundation for identifying argument structures.

One of the primary tasks in AM is identifying arguments, often referred to as Argument discourse Units (ADU) or non-ADUs, when the discourse lacks an argumentative structure. The concept of ADUs originates from the simpler notion of Elementary Discourse Units (EDU). Although EDUs are basic units, two EDUs can be logically connected to form a single ADU, thereby creating a complete argument [Peldszus, 2014]. While ADUs are widely used in AM research, the absence of a unified framework for AM tasks has led to variations in their definition. This divergence reflects the complexity of argument structures, as developing a data model capable of accommodating all possible variations is a significant challenge. For example, enthymemes, a concept introduced by Hitchcock (1985), exemplify a phenomenon where arguers rely on implicit assumptions or premises. Although these implicit components do not necessarily weaken the argument, they increase the complexity of identifying ADUs.

AM still faces certain limitations and significant gaps in its state of the art. The next two subsections section explores the applications, purposes, and existing tools associated with AM and debates, addressing its challenges while providing an overview to facilitate a deeper understanding of its current state.

2.3 Argument Mining in Debates

Debates, whether written or spoken, are a form of communication that employs argumentative elements. For this reason, researchers in the field of AM dedicate significant efforts to studying this genre. Much of the research focusing on written debates centers on online platforms, particularly social media. In this context, various classifications tasks are applied, such as categorizing responses into "supports with argument and justification", "supports without justification", or "does not explicitly support". These classifications take into account responses from a tweet and employ models such as Bidirectional Encoder Representations from Transformers (BERT), Convolutional Neural Network (CNN), and Extreme Gradient

Table 1. Research characteristic on AM and debates

Research	Context	Spoken or Written	Micro or Macro	Structure
Lippi and Torroni [2016]	Political	Spoken	Micro	Structured
Duthie <i>et al.</i> [2016]	Political	Spoken	Micro	x
Visser <i>et al.</i> [2019]	Political	Spoken	Macro	Structured
Haddadan <i>et al.</i> [2019]	Political	Spoken	Micro	Structured
Mestre <i>et al.</i> [2021]	Political	Spoken	x	Structured
Bhatti <i>et al.</i> [2021]	Twitter	Written	Micro	Unstructured
Mancini <i>et al.</i> [2022]	Political	Spoken	Micro	Structured
Hautli-Janisz <i>et al.</i> [2022]	Political	Spoken	Macro	Semi-Structured

Table 2. Research on AM and debates and their data model for argumentation

Research	Data Argument Model
Lippi and Torroni [2016]	Evidence (Study, Expert, Anecdotal and Face Value) Claim (Epistemic, Practical and Moral)
Duthie <i>et al.</i> [2016]	Ethos support Ethos attack
Visser <i>et al.</i> [2019]	Illocutionary connections Inferences Conflicts Rephrases
Haddadan <i>et al.</i> [2019]	Claim Premisse
Mestre <i>et al.</i> [2021]	Relation (Support, Attack or Neither)
Bhatti <i>et al.</i> [2021]	Premisse Stance (Pro or Against)
Mancini <i>et al.</i> [2022]	Evidency (Study, Experct, Anecdotal and Facevalue) Claim (Epistemic, Pratical and Moral)
Hautli-Janisz <i>et al.</i> [2022]	Locutions Propositions Propositional relations (Inference, Conflict and Rephrase) Illocutionary relations (Asserting, Agreeing, Arguing, Assertive Questioning, Challenging Disagreeing, Pure Questioning, Restating, Rhetorical Questioning and Default Illocuting)

Boosting (XGBoost) in their tests [Bhatti *et al.*, 2021].

However, social media is not the only source of written debates. Researchers also analyze online discussion forums, applying AM techniques to classify argumentative elements such as “conclusion”, “premise”, “backing” (additional information to support an argument), “rebuttal” (attacking a conclusion), and “refutation” are performed using Support Vector Machine (SVM) [Habernal and Gurevych, 2015]. Other studies highlight additional sources of arguments, such as Wikipedia [Schneider, 2014].

Researches have also developed tools for visualizing and evaluating argumentative data from Questioning and Answering (Q&A) platforms [Carstens *et al.*, 2014]. Furthermore, some studies focus on automatically identifying opinions on controversial topics (abortion, LGBTQIA+ rights, drug legalization and others) by analyzing comments in online forums. These efforts use textual similarity as a primary methodology to determine whether an author supports or opposes a specific issue [Boltužić and Šnajder, 2015]. This current study also focuses on controversial topics, since in the DEBISS

corpus a set of controversial topics was chosen to guide discussions during debate sessions. These topics include the use of LLMs in education, impacts on jobs, data rights, and more.

In this context, it is important to emphasize that debates are not limited to written texts; many occur orally, and researchers have applied AM techniques to oral debates to explore their unique features. Among these, political debates have received considerable attention. Studies in this area focus on identifying conclusions, supportive or attacking arguments, and the relationships between arguments and premises. Models such as SVM, M-ArgNet, and Bidirectional Long Short-Term Memory (BiLSTM) have been employed for these classification tasks [Mancini *et al.*, 2022; Haddadan *et al.*, 2019]. Beyond conventional classifications, some researchers have explored argumentative phrases appealing to “ethos” (credibility), classifying them as either “ethos attack” or “ethos support” in the context of political debates [Duthie *et al.*, 2016]. Also, studies have investigated the classification of claims, such as those extracted from the UK political election debates in 2015, focusing specifically on the classification of

argument components like claims [Lippi and Torroni, 2016].

Further studies classify argumentative elements in oral and political debates, considering not only the text from transcription but also audio features by applying a multi-modal model [Mestre *et al.*, 2021]. The *M-Arg* dataset was built using crowd-sourcing techniques, models such as CNN, and BiLSTM have been employed to classify arguments while incorporating both textual and audio representations. Additionally, some studies present comparative results between models that use only text versus those combining audio and text, finding that text-only models based on BERT often achieve superior performance [Mestre *et al.*, 2023].

Despite advancements, AM research on debates remains largely concentrated on political debates, social media platforms, and online forums. Other oral genres, such as those in legal contexts, educational environments, TV shows, and interviews, have received minimal attention. However, these genres represent rich sources of argumentation, deserving of greater attention and investigation in AM. It is also important to emphasize that the structure of political debates, forums, and social media discussions tends to follow a very **structured** and limited format, with strict rules for time and turn-taking. Such formats may not be suitable for other debate contexts and scenarios.

Since the application of AM in debates can be broad, ranging from detecting micro and macro relations to identifying ADUs and their components, Table 1 provides a detailed description of various AM studies and their respective data representation formats. This highlights the wide applicability of AM tasks in debates and draws attention to certain tasks that are not yet covered, shedding light on existing research gaps.

For a better understanding of the overall state of AM and debate research, refer to Table 2. It provides details on the data annotation methodology and indicates whether the debates analyzed in the research are structured or unstructured. One of the main objectives of this research is to analyze and address the challenges faced by AM in processing debates that deviate from the rigid structures typical of political debates. The proposed DEBISS dataset exemplifies this by featuring debates free from time constraints and fixed numbers of speaking rounds. This format can often be predominant in educational settings, such as classroom activities or discussion panels, or even daily basis activities.

2.4 Argument Mining And Large Language Models

Since 2019, LLMs have advanced rapidly, becoming essential tools in numerous NLP tasks across various scenarios. Their application to AM is no exception. However, research in this area remains in its early stages, with the integration of LLMs into AM still underdeveloped in many contexts. This subsection aims to explore and better understand the potential of LLMs in AM.

The research by Van der Meer *et al.* [2024] utilized LLMs to generate and summarize new arguments on specific topics by classifying ADUs and stances. This study focused on arguments derived from public opinions about COVID-19 health safety measures. By combining manual and automated

methods for argument generation, the approach achieved results comparable to fully manual efforts while significantly enhancing time efficiency.

Similarly, [Rajasekharan *et al.*, 2023] proposed an approach for automatically generating AM data using LLMs, focusing on texts related to the Malaysian Airlines flight MH17 tragedy. This method employs LLMs to identify the main claim and supporting evidence. Based on the main claim, new sub-claims are generated using a specific framework with models called FrameNet, a human-curated lexical database. This step addresses the challenges faced by other LLMs in generating sub-claims effectively. For each sub-claim it is searched for supporting evidence, producing sufficient data to create a system capable of automatically proving claims through text generation with LLMs. Although the initial results are promising, the study does not detail the evaluation processes and is presented as a preliminary effort. Future research aims to test this approach on more complex texts to better assess its effectiveness [Rajasekharan *et al.*, 2023].

Going beyond data generation, there are studies that explore argument quality evaluation using LLMs. For instance, in van der Meer *et al.* [2022] study, the focus is on extracting two key aspects of arguments: validity and novelty. Validity ensures a logical connection between premises and conclusions, while novelty introduces new information to the argument. These metrics are critical for evaluating argument quality. The study employs ChatGPT-3 with prompt engineering techniques and incorporates a Multi-Task Learning approach, where auxiliary models are fine-tuned on intermediate tasks before results are passed to the final model. For each task - validity and novelty - custom prompts, designed following prior research guidelines, enhanced task performance. [Alivanistos *et al.*, 2022]. Among the tested approaches, the best results achieved were an F1 score of 75% for validity and 62% for novelty.

Correspondingly Mirzakhmedova *et al.* [2024] investigates using LLMs, particularly GPT and Pathways Language Model models, for argument quality analysis. The research compares the annotations provided by LLMs with those of novice and expert human annotators, concluding that LLMs demonstrate close agreement with human annotations and greater consistency compared to human labeling efforts. Also, it is important to highlight that the quality of text argumentation in this research is classified into three levels: high, medium, or low.

Due to the different amounts of multiple LLM available for use, many researchers try to create a benchmark on their research to compare the results on different AM tasks with different LLM. To this end, a research focused on compiling fourteen different AM corpora and testing multiple AM tasks on several LLMs [Chen *et al.*, 2024] and creating a comparison. The tasks included stance, evidence and claim. The study investigated models such as the GPT-3.5-Turbo, Flan, and Large Language Model Meta AI (LLaMA) 7B models. The results indicated strong performance from the GPT-3.5-Turbo and Flan models, while the LLaMA models often underperformed, sometimes below a random baseline. Among GPT-3.5-Turbo and Flan, the latter consistently exhibited greater proficiency in tasks such as claim and evidence detection, as well as stance detection, particularly in binary classifica-

tion scenarios. However, the performance of Flan models declined notably in tasks requiring classification across more than two classes.

In addition, the research proposed by Chen *et al.* [2024] highlights the benefits of a few shot techniques, revealing that the addition of just ten examples can significantly improve model performance. Although the study presents valuable insights, it is limited by the absence of the latest state-of-the-art models with higher precision and does not explore the potential for performance improvement through fine-tuning approaches.

An indirect application of LLMs in AM task is demonstrated by Rocha *et al.* [2023]. The research focuses on the process of incorporating discourse markers (words that play a crucial role in argumentation) into texts using LLMs. In this study, the authors investigate how the inclusion of these markers influences the AM process. Although the research does not directly apply LLMs to AM tasks, it demonstrates their potential to enhance the AM process. Tests conducted on three distinct datasets, including hotel reviews, micro texts, and essays, demonstrated positive results when incorporating discourse markers, utilizing LLM models such as ChatGPT and Bard as the primary tools in the experiments [Rocha *et al.*, 2023].

A common application of AM involves analyzing legal texts. For this reason, researchers have explored approaches using GPT-3.5 and GPT-4 to identify premises and conclusions in such texts [Abdullah *et al.*, 2023]. These studies compared the performance of both models using a few-shot approach in prompt engineering. The methodology employed direct text classification, where the LLM categorized text into predefined classes based on specific definitions. The results showed an F1 score of 85% for premises and 52% for conclusions when using GPT-4 in combination with a secondary embedding model. This approach underscores the potential of LLMs for precise classification tasks within legal argument mining. However, instead of following the commonly adopted direct text classification methods, the experiment proposed in this research takes an alternative approach by employing text zoning, similarly to the entity recognition process, but applied to sentences instead of tokens. This method shifts the focus from classifying a given text into predefined categories to directly extracting text segments that correspond to specific classes within the text.

Efforts to extend AM beyond written texts have begun to emerge, with researchers investigating the use of LLMs such as GPT-4, for the automatic analysis of arguments in podcasts [Pojoni *et al.*, 2023]. This research applies a strategy to categorize elements such as claims, premises, stances, main claims, counter arguments and rebuttal within transcribed podcast text. However, the validation of results in this study relies on manual inspection by a single expert, which raises concerns about the reliability of the results. The authors emphasize the need for further exploration of AM in podcasts and acknowledge that their study remains in its early stages, requiring significant development, particularly in the area of prompt engineering, which was not extensively addressed in the methodology. This highlights the importance of designing and testing diverse prompt engineering approaches, which is one of the objectives of this proposed study in this research.

It is noteworthy that there are significant gaps in the state of the art concerning AM and LLMs. As both fields are relatively new, many studies are still emerging, with the majority of significant contributions published recently, primarily in 2023 and 2024. Furthermore, several of these studies were published in parallel with the development of this research. These factors highlight the need for further exploration and emphasizes the importance of advancing the state of the art in AM, particularly in the context debates, such as those proposed in the DEBISS Corpus.

3 DEBISS Corpus

The DEBISS corpus consists of audio transcriptions from in-person debates organized with the consent of college student participants in their first year of a computer science under graduation, and a moderator who managed the debate sessions. The data was collected in 2024, consisting of 67 students who formed 16 debate groups, resulting in 9 hours and 35 minutes of audio recordings. These recordings were later manually transcribed by using an semi automated approach, with speech to text models and manual human validation.

The moderator is responsible to explain the rules of the debate, maintaining order during the discussions, promoting interaction among participants, asking targeted questions, and ensuring that everyone had the opportunity to speak without interruptions. With that in mind, it is important to highlight that the debates are **semi-structured**, combining predefined mandatory questions and moments of unrestricted expression. This format allows participants to freely express their viewpoints, respond, and counter-argue at any time, provided they do not interrupt or speak over others. Additionally, the debates follow a basic structure with various stages, including rounds of directed questions. For these reasons, the debates are considered ISS.

Going further, the debate structure adheres to specific rules and a central thematic subject chosen to provoke discussion (“Generative AI and its impacts on society”). Managed by a moderator to maintain order and enhance interactions, the debate is organized into three parts. Moreover, the structure is designed to gradually refine and focus the discussion, guiding participants through a process that deepens their engagement with the topic.

Initially, the moderator explains the rules: debaters must listen to the moderator, avoid interrupting each other, and raise their hands to speak. In the **first part**, debaters present their initial and overall opinions on the topic. During the **second part**, debaters respond to specific questions read aloud by the moderator and displayed on a television, with the opportunity for comments or additional questions from other debaters. This semi-structured interaction facilitates a broad exchange of ideas and encourages engagement among debaters. The **third part** features a final question addressed to all participants, allowing for optional contributions, followed by mandatory final reflections where participants consider their opinions after the debate. In this final moment, debaters are directly asked whether they maintain their initial position. This final reflection not only allows participants to assess how their perspectives have been influenced by the debate

Table 3. DEBISS-Arg labels

Label	Type
ADU, Premise, Claim, and Evidence (example, expert opinion, history, context, debater citation, data)	Plain Text
Support and Attack	Relation Label
	Micro Lever
Questioning, Agree, Disagree, Argue and Partial Disagree	Relation Label
	Macro Level

but also provides valuable insights into the effectiveness of the discussion in challenging their viewpoints.

This corpus was designed not only to address the needs of AM tasks but also to provide rich annotations applicable to various NLP tasks. It represents a collaborative effort by postgraduate and undergraduate students who formed a research group. The group’s primary focus was to develop a corpus that advances the state of the art in NLP for Brazilian Portuguese, specifically in the context of debates.

The corpus includes annotations for disfluency detection (DEBISS-Disfluency¹) [Lima and Campelo, 2024], debater quality analysis (DEBISS-Eval²), and AM tasks (DEBISS-Arg³), which will be detailed in the following subsection. The corpus can also be highly useful for speech-to-text tasks, voice print and speaker diarisation, providing a significant advancement in the availability of open data for NLP tasks in **Brazilian Portuguese**.

3.1 DEBISS-Arg Corpus

The DEBISS-Arg corpus is used in this research as a means to advance the state of the art in AM and debates with LLM. This fully labeled corpus was created from scratch to enable the testing of various AM tasks. Among these tasks, the corpus includes annotations for ADUs and non-ADU text, as well as premise, claim, and evidence components. Additionally, it includes relation annotations between these components, with annotations at the micro level (among components within a single ADU) and the macro level (across different speech utterances in the debate, mapping interactions among the debaters), for further details on the annotation labels presented on the DEBISS-Arg check Table 3.

These sets of labels make the corpus highly comprehensive and suitable for numerous AM tasks. To achieve the final version of the corpus, a detailed methodology for data collection and processing was developed, adhering to specific guidelines throughout the creation process. This section summarizes the methodological steps designed and implemented in developing the DEBISS-Arg corpus.

The creation of an AM corpus is not an easy task. Many research during the creation process reports the complexity to create AM corpus in its many variations when talking about contexts, text classes used, methodology, and so on. The complexity lies in the fact that there is no universal perspective for these models and methods, which complicates the selection of argumentation models for new intelligent software systems

[Walton *et al.*, 2008; van Eemeren *et al.*, 2014]. Efforts to formalize and compile studies on argumentation schemes can be found in the literature, and these efforts are not new [Walton *et al.*, 2008], but there remains a challenge due to a lack of consensus.

The first challenge on creating an AM corpus is the formalization of the annotation protocol, since there is no universal adopted formalization. The primary step in the methodology for AM data creation is the establishment of the annotation protocol that is suitable for the specific context (in this case DEBISS-Arg corpus). The main goal of the protocol is to conduct an in-depth AM analysis of debates close to the DEBISS corpus format, embracing the diverse characteristics present in these kind of debates. It is important to note that the protocol does not propose a new argumentation model or schema. Instead, it consolidates several definitions that are validated in the process of AM by researchers in various scenarios and applications and are adequate for the proposed context. It leverages the strengths of these models to adapt them for the context of debates in the DEBISS-Arg format. To build the proposed protocol, it was necessary to conduct an in-depth investigation of the existing models, aiming to identify the theoretical aspects on which those models are based, as well as the methodology for data annotation that is used. The details regarding the protocol definition and data annotation process are available in the Github page. Additionally, the complete protocol in PDF format can be accessed via the GitHub repository⁴.

As previously mentioned, a specific protocol was developed to process transcriptions from debate sessions. Once applied, this protocol produces a transcription text annotated with multiple layers of information for AM. The protocol was shared with professional annotators, this process was conducted by two experienced language data analysts with backgrounds in data labeling for NLP tasks. They manually annotated the data and performed a pairwise review to ensure consistency and quality.

Figure 1 presents a simplified example illustrating how the final data appears after being processed by professional annotators. Additionally, the fully annotated corpus is publicly available in the GitHub repository⁵, where it can be accessed and used for research related to AM.

4 Methodology

This section presents the methodological approach adopted in this research. The first subsection discusses the experiments conducted to automatically extract argument structures. This process is carried out using prompt engineering techniques, with the DEBISS-Arg corpus serving as a source for gathering metrics. Additionally, this section details the evaluation criteria and metrics used to assess the effectiveness of LLMs in AM tasks.

¹<https://github.com/AINDA-Project-UFCG/disfluency-data>

²<https://github.com/AINDA-Project-UFCG/debater-quality-data>

³<https://github.com/AINDA-Project-UFCG/argument-mining-data>

⁴<https://github.com/AINDA-Project-UFCG/argument-mining-data/blob/main/protocol.pdf>

⁵<https://github.com/AINDA-Project-UFCG/argument-mining-data>

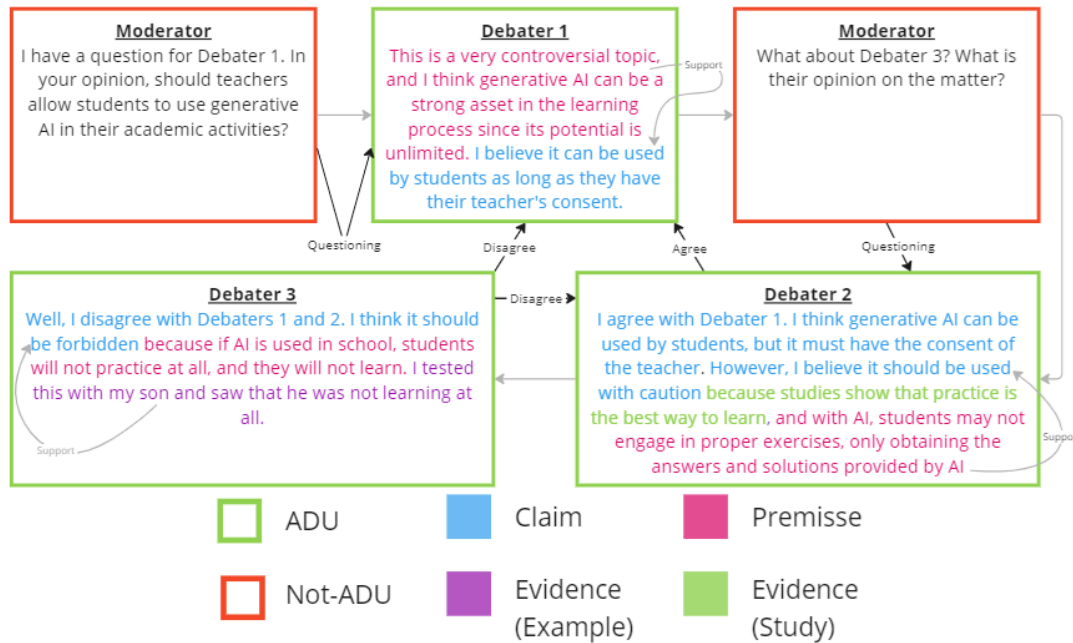


Figure 1. Protocol usage illustration of a simplified debate

4.1 Prompt Engineering

The prompt engineering process is fundamental in LLM research, as the techniques applied can yield different results across various NLP tasks [Marvin *et al.*, 2024]. Prompts influence how LLMs interact with text, significantly impacting the obtained results. Therefore, investigating and applying prompt engineering techniques is of great importance, as it directly affects the research outcomes.

To generate the analysis data from the DEBISS-Arg corpus, the tested prompts were designed to mimic the behavior of human annotators in the manual annotation process. The proposed prompt specified the input, which consisted of a text transcription from a single speaker in the debates. In summary, the LLM was instructed to highlight text spans that represent arguments and their components and put those text span in an JavaScript Object Notation (JSON) format as an output. Figure 2 provides an example of one of the prompts used in this process. Also, all the proposed prompts are crafted using the best practices recommended in the OpenAI documentation⁶.

To find the most effective prompt, several attempts were made. The best results were achieved by applying two prompt engineering techniques: few-shot prompting and chain-of-thought (CoT) reasoning. After testing each technique individually and later combining them, the latter approach yielded the best performance. Consequently, all developed prompts incorporated both techniques.

Three different prompting strategies were designed and tested, as outlined below:

- **Single-Prompt Strategy:** Using a single prompt to execute the entire process.
- **2-Prompts Strategy:** Dividing the task into two prompts—one for identifying arguments and another for identifying argument components.

- **4-Prompts Strategy:** Utilizing four specialized prompts to handle specific sub tasks — one for arguments and three others, each dedicated to a specific argument component.

In the **Single-Prompt Strategy**, the approach involved providing a transcription from an utterance in the debate and requesting the LLMs to conduct a full analysis of the text, retrieving all arguments and their components at once. For more details, refer to Figure 2 that has a prompt example.

Moreover, the **2-Prompts Strategy** was developed to improve results. This approach, similar to single prompt strategy, was implemented using two distinct prompts. The first prompt is responsible solely for identifying arguments in a given transcription text. In other words, it performs a single task rather than simultaneously searching for both arguments and their components. If the first prompt detects an argument, a second prompt is executed to identify the three types of argument components within the text that has an argument. However, if no argument is found by the first prompt, the second prompt is never executed. To better understand this approach, refer to Figure 3 that has a flow chart for the 2-prompts strategy, for more details on the prompts the reader can check Appendix B.

The **4-Prompts Strategy** is built upon the 2-Prompts Strategy. However, instead of identifying all argument components at once in a single prompt, each component type (claim, premise, evidence) has its own specialized prompt. As in the previous strategy, argument components are only analyzed if an argument is first detected in the text. For more details on this approach, refer to Figure 4 containing a flow chart for the 4-prompts strategy and Appendix C that has details on the prompts texts.

⁶<https://platform.openai.com/docs/guides/prompt-engineering>

You are a proficient debate analyst capable of identifying and analyzing argumentative aspects in debate transcriptions.

Given an audio transcription text from a debate, delimited by triple backticks, about 'Generative AI and its impact on society,' your objective is to analyze the argumentative aspects of the text following the protocol described below. The response must be a valid JSON file without any prose.

Follow these steps to analyze the transcription debates:

1 - Identify the text spans of arguments. Some speeches might not have an argument, while others might have more than one. Using the following definition below:

- Argument: "some argument definition here"

2 - If you do not find any argument in the text, return a JSON in the format with blank strings as in the example below:

```
{
  "argument": [],
  "claim": [],
  "premise": [],
  "evidence": []
}
```

3 - If you find arguments, for each argument get the text from the argument and look for the argument's components using the definitions below. Every argument must have a claim and at least one evidence or premise.

- Premise: "some premise definition here"
- Claim: "some claim definition here"
- Evidence: "some evidence definition here"

4 - Do not rewrite what's in the original text; it must be the same text from the input in the output. (Copy and paste the span of text that represents an argument or argumentative component)

5 - Place the results in the output as described below in the output format. The output format must be a JSON file with the following structure. Only generate valid JSON and no prose:

```
{
  "argument": ["span of text with argument 1", "span of text with argument 2"],
  "claim": ["span of text with claim 1", "span of text with claim 2"],
  "premise": ["span of text with premise 1", "span of text with premise 2"],
  "evidence": ["span of text with evidence 1", "span of text with evidence 2"]
}
```

6 - Follow the examples input and output below:

"some examples with input and output"

```utterance trasncription text```

Figure 2. Prompt example (CoT + Few-shots)

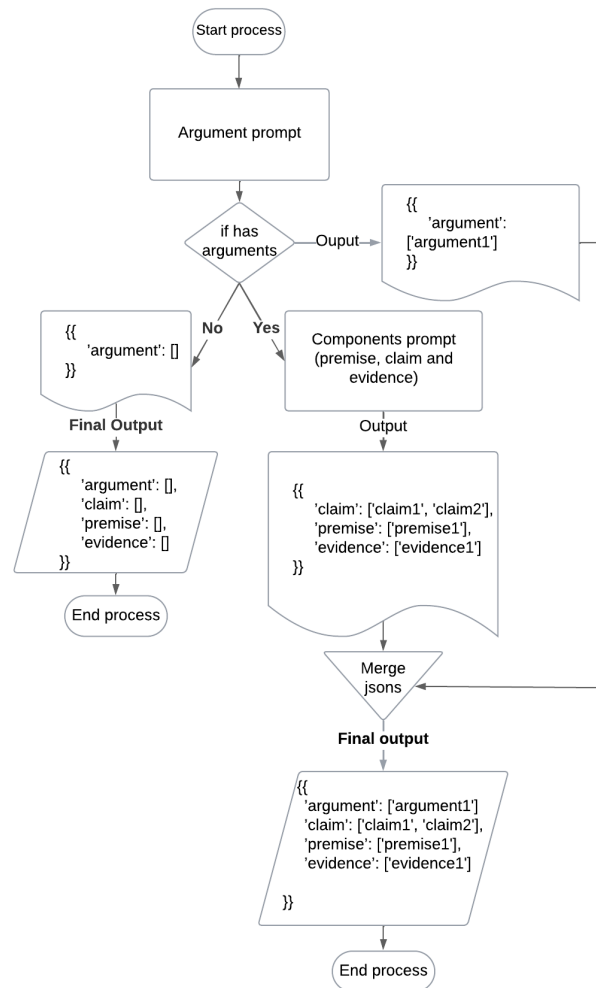


Figure 3. 2-Prompts strategy

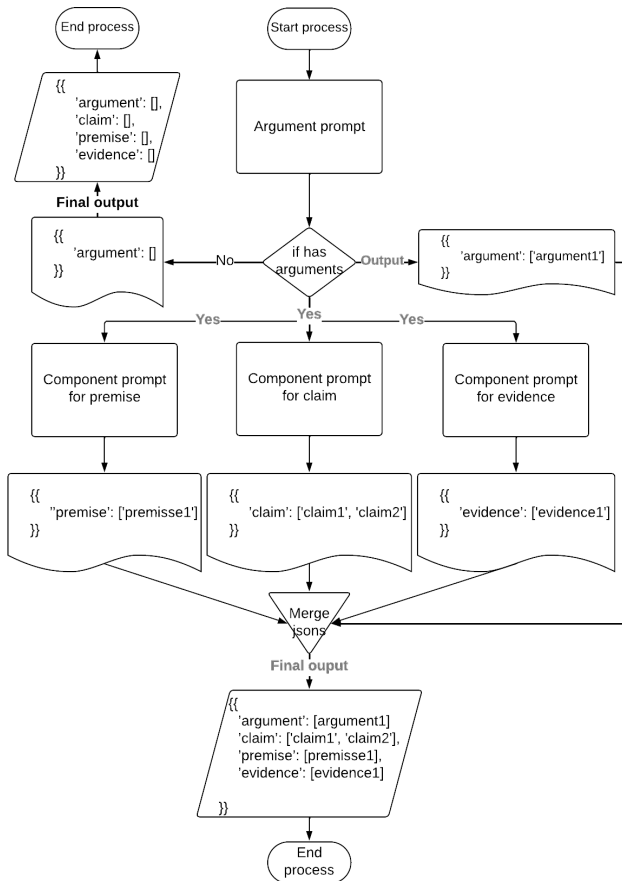


Figure 4. 4-Prompts strategy

## 4.2 Evaluation Metrics

To evaluate the proposed experiments, it is necessary to design specific metrics tailored to the characteristics and objectives of this research. Common NLP metrics such as accuracy, precision, recall, and F1 score are widely used because they provide valuable insights into results, particularly for direct text classification tasks. For instance, when a given text belongs to a predefined class, an AI model's inference is compared to manually labeled data (ground truth), making these metrics suitable and effective.

However, in this research, direct text classification is not used to assess the LLMs' ability to extract argumentative data. Instead, the experiments involve feeding the LLM with unstructured text and expecting a structured output. In this context, the structured output consists of AM data labeled text. This approach deviates from typical text classification tasks and aligns more closely with text zoning processes, where the objective is to extract text spans and corresponding labels rather than assigning labels to the text alone, which is very close to the Entity Recognition (ER) approach. For this reason, custom metrics are required to analyze the results effectively in this experimental set up.

The goal is to evaluate how accurately the LLM can extract text spans that represent specific AM labels from a given text. This involves assessing how closely the LLMs' annotations mimic those of human annotators, presented in DEBISS-Arg corpus. To achieve this, text similarity techniques are employed as a means of measuring the LLM's accuracy. Specifi-

cally, the approach relies on calculating the similarity between the text spans output by the LLM and those annotated by human annotators. For this purpose, the Levenshtein distance is used as a mean for generating the metrics.

The **Levenshtein metric** introduced by Levenshtein (1966) measures the similarity between two words by calculating an edit distance. Text similarity is a crucial concept in NLP and information retrieval, often used to quantify how closely two textual elements are similar to each other. The Levenshtein distance, also known as edit distance, it calculates the minimum number of character edits — insertions, deletions, or substitutions — required to transform one string into another.

Hence, the resulting metric provides a robust measure of dissimilarity, with a lower distance indicating higher similarity. This technique is particularly effective for handling typographical errors, detecting plagiarism [Soyusiawaty and Rahmawanto, 2018], and comparing sequences in biological data. Researchers derive a similarity score between 0 and 1, where 1 represents identical strings. This normalized metric enables the analysis of diverse datasets. The normalized Levenshtein metric plays a crucial role in this research methodology and is a key aspect of evaluating LLM performance on the proposed AM tasks.

To generate metrics for the experiments based on Levenshtein distance, an evaluation algorithm was developed. The goal of this algorithm is to compare the LLM output with the ground truth annotations from manual labeling and generate metrics to analyze the LLMs' efficacy. The algorithm is divided into four cases (Figure 5):

- Case 1: Both the LLM output and manual annotations are empty. This indicates the text contains no arguments or components.
- Case 2: The LLM produces more text spans than the ground truth annotations, suggesting the LLM may have added extra and incorrect text spans.
- Case 3: The LLM produces fewer text spans than the ground truth annotations, indicating it may have missed some spans.
- Case 4: Both the LLM and ground truth annotations have the same number of text spans, which could imply that the LLM successfully captured the same data as the manual labels.

However, it is important to emphasize that simply comparing the number of text spans is not sufficient to evaluate quality. The next step in the algorithm involves comparing the similarity of the text spans themselves. This ensures that the LLM accurately captures the correct text spans that correspond to each label during the text zoning process.

In the algorithm's next step, after identifying the cases, text similarity metrics based on Levenshtein distance are generated. The algorithm compares all LLM text spans outputs with all the ground truth annotations, by forming all possible pairs of text spans from both groups and calculating the Levenshtein metric for each pair. The pair with the highest similarity score is selected to represent the final data for results. To enhance the readability and understanding of the metrics, the approach includes calculating the following items, for more details refer to Figure 5:

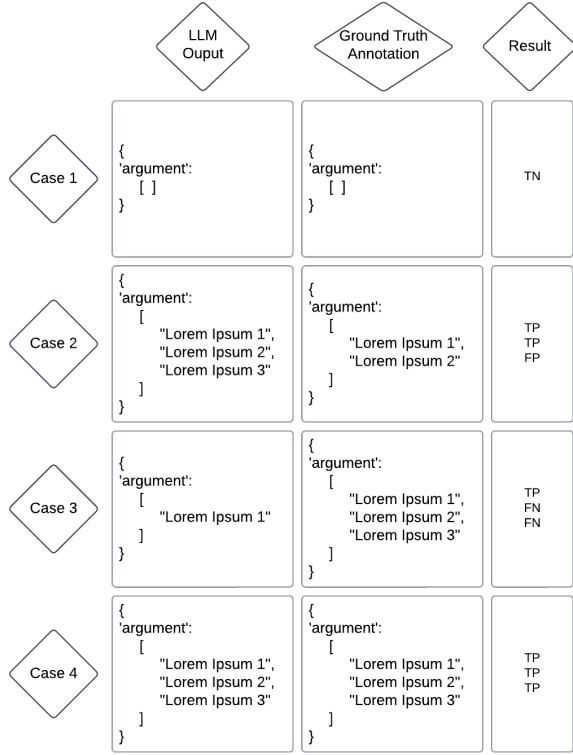


Figure 5. The algorithm cases

- First case: where the LLM output and ground truth annotation are empty, 1 True Negative (TN) is produced.
- Second case: where the LLM provides fewer text spans than ground truth annotation, it produces 2 True Positives (TP) and 1 False Positive (FP).
- Third case, where the LLM provides more text spans than the ground truth annotation it produces 1 TP and 2 False Negatives (FN).
- Fourth case: where the LLM and ground truth annotation sizes match, 3 TP are produced.

It is important to emphasize that a TP is only considered when the Levenshtein similarity between a pair of text spans is greater than 70%. This specific value is chosen to better demonstrate the inefficiency of LLMs in the proposed tasks by setting a low threshold. Even though, with the low threshold the results were not promising, hence they further highlighted the limitations of LLMs in AM tasks using the proposed methodology.

Also, setting this threshold is crucial, as the goal is not just to count the number of text spans but to assess the similarity between them. By focusing on similarity, the algorithm ensures that only those text spans that closely match in content are considered accurate, providing a more reliable evaluation of the LLMs' performance. There for, with the counts of TN, TP, FP, and FN cases, it is possible to generate the classical text classification metrics, which will provide valuable insights into the results. These metrics will help in better understanding the LLMs' performance. Additionally, the Levenshtein similarity values and the case counts will assist in evaluating how accurately the LLM captures argument components, offering a more comprehensive view of its effectiveness.

## 5 Results

This section presents the results obtained by applying the proposed methodology, establishing a benchmark among the tested LLMs across various AM tasks.

### 5.1 Initial Tests And Prompt Engineering

The first step in this methodology is to determine the most effective prompt engineering technique before the creation of the the proposed LLM benchmark. To achieve this, three distinct strategies were adopted to identify the most suitable solution for AM tasks. These strategies, along with their results, are detailed in this subsection. It is important to note that the results of these initial tests were based exclusively on a single debate (Debate 1) to mitigate the possibility of the LLM having prior exposure to the DEBISS-Arg data and to ensure a static benchmark. To evaluate the three different strategies tested in this phase, the metrics outlined in Section 4.2 were applied. This approach aimed to provide a more reliable assessment of the prompts and strategies, ensuring robustness in the evaluation process.

During the initial testing phase, GPT-3.5 Turbo was used for evaluation. Three different prompt strategies were proposed. The first was the single-prompt strategy, in which four different prompts with distinct prompting techniques were tested. The second was the 2-prompts strategy, and the third was the 4-prompts strategy.

The first prompt employed in the single prompt strategy was the most basic. The prompt involved providing a transcription of an utterance from the debate and requesting the LLM to perform a comprehensive analysis of the text, retrieving arguments and their components simultaneously. The F1 scores for the various components of the arguments were as follows: arguments scored 30%, premises scored 6%, claims 7% , and evidence was not available, indicating that no TP were identified for evidence detection task, its is assigned NA values for those cases.

After analyzing the first prompt results, a new prompt technique was adopted to improve performance. This time, a few-shot prompting technique was implemented using the same base prompt but incorporating a few examples. The resulting F1 scores for the second prompt were 84% for arguments, 20% for premises, 24% for claims, and 20% for evidence.

Building on this, the third approach introduced CoT reasoning, simulating the annotation process in five steps. This method was inspired by the same protocol used by professional annotators in their manual annotation process. The resulting F1 scores were 26% for arguments, 14% for premises, 11% for claims, and NA for evidence, since no TP were found.

The fourth prompt used was the same as the third but included a few examples to guide the model in the CoT process. The resulting F1 scores were 79% for arguments, 22% for premises, 30% for claims, and 29% for evidence. Since these initial testing showed that combining few-shot prompting and CoT reasoning achieved the best results, the prompts used in the subsequent 2-prompts and 4-prompts strategies were based on these two techniques.

In the 2-prompts strategy, the task was split into two distinct

**Table 4.** F1 Score For AM Tasks and Each Prompt Approach (Debate 1 Only)

Arg = Argument; Pre = Premise; Cla = Claim; Evi = Evidence; Avg = Average

	Arg	Pre	Cla	Evi	Ave
<b>Basic Prompt</b>	30%	6%	7%	NA	10.8%
<b>Basic + Few-shots</b>	84%	20%	24%	20%	37%
<b>CoT</b>	26%	14%	11%	NA	12.8%
<b>CoT + Few-shots</b>	79%	22%	30%	29%	40%
<b>2-Prompts Strategy</b>	85%	11%	13%	20%	17.3%
<b>4-Prompts Strategy</b>	86%	17%	29%	27%	39.7%

**Table 5.** F1 score average for CoT + few-shots and 4-prompts strategy across all debate sections

CoT + Few-shots	4-Prompts Strategy
34.78%	36.53%

prompts. The first prompt identified arguments within the text. If arguments were found, the second prompt identified the argument components, including premises, claims, and evidence. The results for Debate 1 showed F1 scores of 85% for arguments, 11% for premises, 13% for claims, and 20% for evidence.

In the 4-prompts strategy, each argument component, including premises, claims, and evidence, had its own dedicated prompt, further refining task specialization. The achieved F1 scores for Debate 1 were 86% for arguments, 17% for premises, 29% for claims, and 27 % for evidence. These results highlight the potential benefits of further task specialization in argument mining tasks, as each prompt focuses exclusively on a single component, leading to improved performance.

In conclusion, as observed in Table 4, it is evident that the **CoT + Few-shots** and **4-Prompts Strategy** yielded the best results for the Debate 1 section. These findings reinforces the benefits of incorporating examples (few-shot learning) and using CoT techniques to improve outcomes. Given the minimal performance difference of only 0.25% between CoT + Few-shots and 4-Prompts Strategy, further tests were carried out in other debate sections to obtain a more accurate and comprehensive view of the most effective approach. Hence, the results reveal that while the CoT + Few-shots and 4-Prompts Strategy performed closely, the 4-Prompts strategy consistently demonstrated slightly better results and higher average (Table 5), answering RQ4. For this reason, the 4-Prompts strategy was adopted for the remaining experimental setups as in the LLM benchmarking.

Also, it is important to emphasize that the process of executing the initial inference tests using GPT-3.5 for all debates was conducted within the same week. Additionally, the further testing required for the benchmark establishment, with the other LLMs, was also performed within this same time frame. This reinforces the importance of maintaining a static benchmark, ensuring consistency in the evaluation process and minimizing potential variations in model behavior over time.

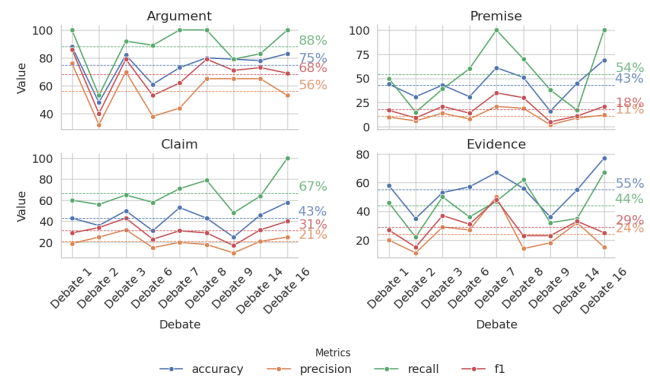
## 5.2 LLMs Benchmark

To address RQ 1 and 2, an experimental setup was designed to create a benchmark across the four proposed AM tasks. The LLMs used in this process included ChatGPT-3.5 Turbo, ChatGPT-4, Gemini, and LLaMA 70B and *Sabiá* 3. Each subsection below presents the results for each LLM. It is important to emphasize that this phase of the research was conducted using the 4-Prompts strategy and was carried out across all nine available different debate sections.

### 5.2.1 ChatGPT-3.5 Turbo

The ChatGPT-3.5 Turbo model demonstrates strong performance in the argument classification task. With a high recall of 88%, it excels at identifying positive instances, minimizing false negatives. However, its precision is relatively low at 56%, meaning it generates many false positive predictions. The F1 score of 68% indicates a moderate balance between precision and recall, suggesting that while recall is prioritized, the model's precision still needs improvement.

Furthermore, when evaluating other AM tasks such as premises, claims, and evidence, the results reveal that the model struggles with these tasks. In the case of premises, the model shows several weaknesses, with a recall of 54%, indicating it detects some positives but with a very low precision of 11%, meaning many positive predictions are incorrect. Its accuracy of 43% and F1 score of 31% reflect poor overall performance, suggesting the model has difficulty with correct classifications. Examining the output examples, it is clear that the LLM generates a lot of false positives text spans, by classifying texts that should not be premises. Going further a similar behavior can be seeing in the claim and evidence classification. Check Figure 6 to see all the metrics for each debate section, as well as the average values for these metrics.

**Figure 6.** Performance metrics across debate sections with averages - ChatGPT-3.5 Turbo

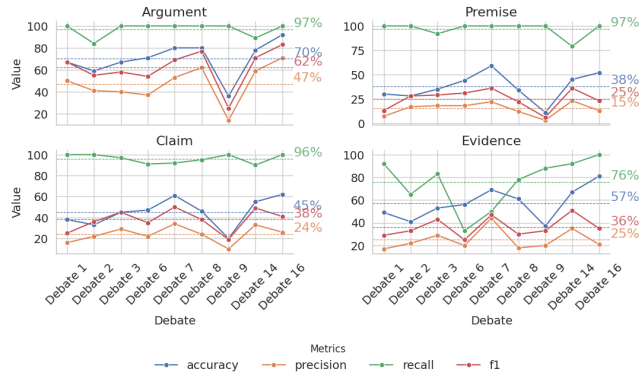
### 5.2.2 ChatGPT-4

For ChatGPT-4, the model's metrics for argument classification show a recall of 97%, indicating it identifies most positive instances. However, its average precision is only 47%, meaning many positive predictions are incorrect. The accuracy of 70% reflects reasonable overall performance, but the F1 score of 62% suggests a poor balance between precision and recall, with the model prioritizing recall at the expense of precision.



Surprisingly, when compared to the results for ChatGPT-3.5 Turbo, the older version of the model outperforms the latest version in the argument task classification.

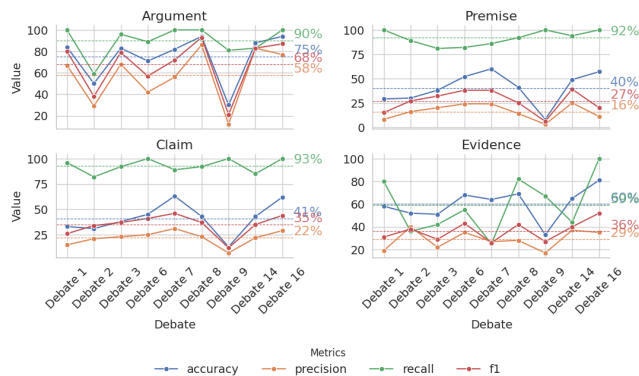
Further analysis of the others classification tasks reveals that while the metrics are still generally low, ChatGPT-4 shows slightly better results in component classification compared to ChatGPT-3.5 Turbo, especially when comparing F1 scores. However, it is worth noting that the model has a high recall but low precision, similar to ChatGPT-3.5 Turbo. This indicates that it generates many false positive cases, suggesting that it is identifying more text spans than it should. Check Figure 7 to see all the metrics for each debate section, as well as the average values for these metrics.



**Figure 7.** Performance metrics across debate sections with averages - ChatGPT 4

### 5.2.3 Gemini

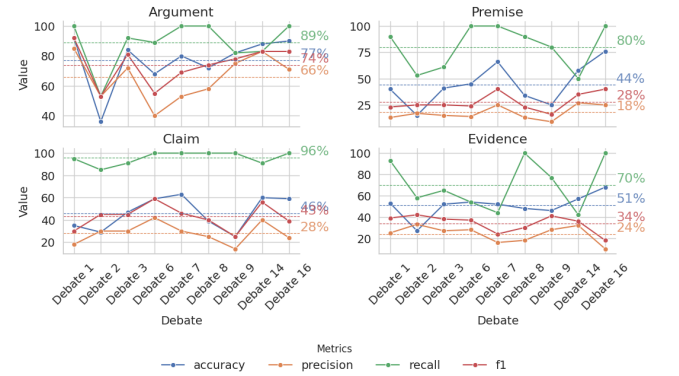
The Gemini model has shown a similar behavior to GPT-3.5 results in the proposed benchmark, with an average F1 score of 68% in the argument classification task. However, when it comes to other classification tasks, such as premises, claims, and evidence, the model's performance falls short in comparison to argument classification. Its F1 score remains below 50%, with high recall values and low precision. This suggests that the model is generating more text spans than it should, similar to the issues observed in the other models. Check Figure 8 to see all the metrics for each debate section, as well as the average values for these metrics.



**Figure 8.** Performance metrics across debate sections with averages - Gemini

### 5.2.4 LLaMA 70B

The LLaMA model performed surprisingly well in the argument classification task, achieving the highest average precision and F1 score (74%). However, when it comes to the other classification tasks, it shows low metrics similar to the other LLMs, facing the same issues as observed in previous models. It is surprising that LLaMA, an open-source model, demonstrated comparable performance despite being smaller in instruction size and problem-solving capability, as reported in multiple NLP tasks. Nevertheless, it still delivered competitive results when compared to the other closed-source and paid models that were tested earlier. Check Figure 9 to see all the metrics for each debate section, as well as the average values for these metrics.



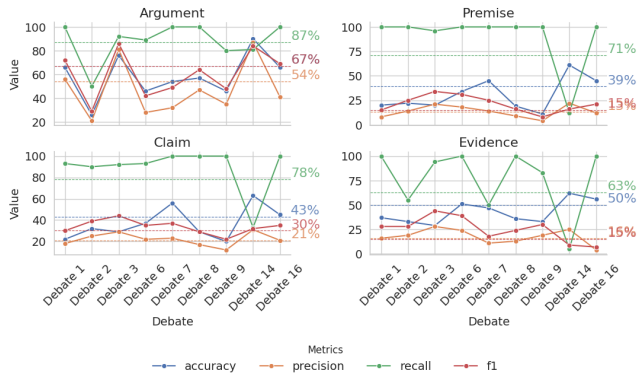
**Figure 9.** Performance metrics across debate sections with averages - LLaMA 70B

### 5.2.5 Sabiá 3

The *Sabiá* model produced results similar to other models when comparing argument and claim identification. However, its performance in premise and evidence detection was even lower, resulting in weaker F1 scores. It appears that training the model in the same language as the dataset does not necessarily yield significantly better results compared to multilingual models (addressing RQ3). Once again, a similar pattern was observed, with high recall and low precision, leading to an unbalanced F1 score. The argument identification task achieved an F1 score of 67%, while premises and evidence scored 15%, and claims scored 30%. Overall, the results were very close to those of other LLMs in many tasks, but LLaMA remained the best-performing model when comparing the results. Refer to Figure 10 for a detailed breakdown of the metrics for each debate section, as well as the average values for these metrics.

**Table 6.** Average F1 score for each LLM across each task

	Arg	Prem	Cla	Evi	Ave
GPT 3.5 Turbo	68%	18%	31%	29%	36.5%
GPT 4	62%	25%	38%	<b>36%</b>	40.25%
Gemini	68%	27%	35%	<b>36%</b>	41.5%
<b>Llama 70B</b>	<b>74%</b>	<b>28%</b>	<b>43%</b>	34%	<b>44.75%</b>
<i>Sabiá 3</i>	67%	15%	30%	15%	25.5%

**Figure 10.** Performance metrics across debate sections with averages - *Sabiá 3*

### 5.2.6 LLM Benchmark Results

All the LLMs exhibited similar behavior across all AM tasks, with slight differences in performance. The LLaMA model appeared to be slightly better than the others in most tasks, with the only exception being a 2% lower average F1 score in evidence classification compared to GPT-4 and Gemini. Further, all models displayed high recall in most of the classification tasks but had low precision. They seem to struggle with generating a lot of TN cases, which suggests that the models might not fully understand the tasks and are instead generating random text spans. This issue is particularly evident in the component classification tasks, which are the most challenging. The argument classification task, being the most basic, yielded the best overall results, but the performance still isn't as high as desired. In Table 6, the average F1 score for each LLM and each AM task across all debate sections is presented.

## 6 Conclusion

This study addressed its primary research objectives by comprehensively evaluating the performance on several LLMs and providing valuable insights into advancing AM tasks within Brazilian Portuguese debate data. The resulting benchmark includes an analysis of LLM behavior in advancing AM capabilities but also details the nuanced behavior of these models. Specifically, this benchmark provides an analysis of LLM performance across diverse AM tasks, critically demonstrating the impact of prompt engineering techniques on their efficacy. These findings highlight how LLMs can be used and assessed for identifying complex argument structures.

Additionally, the proposed methodology introduces new challenges beyond conventional text classification approaches commonly explored in existing AM research, by applying

text zoning process for text classification. This novel challenge, combined with the inherent difficulty of AM tasks underscores the difficulties in achieving high performance in the proposed AM tasks by the tested LLMs.

Surprisingly, the LLaMA LLM emerged as the best-performing model for most AM tasks, outperforming or matching others models in key metrics. Even though LLaMA is known as a smaller and less powerful model it demonstrated competitive performance, achieving results that surpassed larger and more advanced LLMs in most AM tasks. Regarding prompt engineering, the CoT, few shots, and 4-Prompts approaches combined yielded the best results, with metrics that were very similar to 2-Prompts approach, though the 4-Prompts approach was slightly better. The use of few-shot techniques also significantly improved performance, helping LLMs to better understand the tasks and produce more accurate results.

In this sense, for the most basic AM tasks (argument detection), the best results were obtained using the LLaMA model, which achieved an F1 score of 74%. However, its performance dropped significantly for the task of claim detection, with an average F1 score of 43%. For premise detection, the results were even lower, averaging an F1 score of 28%. Only in latter task (evidence detection), both Gemini and ChatGPT-4 outperformed LLaMA, reaching F1 scores of 36%.

Furthermore, it is important to highlight that other studies involving LLM, such as GPT, have reported significantly better performance, achieving F1 scores of 76% for claim detection and 51% for evidence detection [Chen *et al.*, 2024]. Although a direct comparison between studies is not feasible due to differences in datasets and methodologies, the discrepancy in results draws attention to how the language, methodology and debate-based data context in this research may have introduced specific challenges for LLM analysis in the proposed AM tasks.

Several factors may have contributed to the poor results observed in this study, as most of the metrics obtained for the evaluated LLMs were unsatisfactory. These outcomes can be attributed to multiple aspects. First, language barriers may have negatively impacted the models' performance. Additionally, working with transcribed speech presents significant challenges—such as disfluencies and informal patterns—which can affect model accuracy. Furthermore, the methodology adopted in this study differs from traditional text classification approaches. Instead of direct classification, it uses a text zoning method similar to entity recognition. Together, these factors help explain the poor performance of the LLMs in the proposed AM tasks. These findings emphasize the ongoing need for the advancement of AM methodologies and reinforce the importance of extending NLP research to languages other than English.

### 6.1 Future Work

This research provides valuable insights into the behavior of LLMs across a group of AM tasks. It highlights the limitations of LLMs in addressing these tasks by presenting metric measurements that detail their performance. The findings reveal mainly unsatisfactory results for most tasks, emphasizing the need for improvement in this area. Therefore, given



these outcomes, it is crucial to acknowledge the limitations of the current study, which can be addressed in future work. Building on the metrics gathered and following the proposed methodology, future research can refine approaches to enhance LLM performance in AM tasks and contribute to advancing the state of the art in this domain.

It is important to acknowledge that the proposed methodology of text zoning may introduce inherent difficulties for the AM process. This approach generates additional instructions for the LLM to follow when compared to direct text classification. For this reason, implementing a direct task classification process using the DEBISS-Arg corpus could provide a valuable investigation into whether the chosen approach influences the results. Moreover, this alternative methodology offers more precise data for comparison with other studies that evaluate the performance of LLMs in AM tasks using different corpora and language.

## Declarations

### Authors' Contributions

DP was the primary contributor and lead writer, responsible for conceptualization, methodology, experiment execution, and writing. DT was responsible for data curation and writing. CC contributed to the conceptualization and supervised the study.

### Competing interests

The authors declare that they have no competing interests.

### Availability of data and materials

The datasets generated and analyzed during the current study are available in an GitHub repository along side with the Corpus its transcription, AM annotation and audio files.<sup>7</sup>

## References

- Abdullah, A. Z., Michael, G., and Jelena, M. (2023). Performance analysis of large language models in the domain of legal argument mining. *Frontiers in Artificial Intelligence*, 6. DOI: 10.3389/frai.2023.1278796.
- Accuosto, P., Neves, M. L., and Saggion, H. (2021). Argumentation mining in scientific literature: From computational linguistics to biomedicine. In *BIR@ECIR*. Available at: <https://frommholz.org/share/bir2021/ceur-ws/paper-03.pdf>.
- Al Khatib, K., Ghosal, T., Hou, Y., de Waard, A., and Freitag, D. (2021). Argument mining for scholarly document processing: Taking stock and looking ahead. In *Proceedings of the Second Workshop on Scholarly Document Processing*, pages 56–65, Online. Association for Computational Linguistics. DOI: 10.18653/v1/2021.sdp-1.7.
- Alivanistos, D., Santamaría, S. B., Cochez, M., Kalo, J. C., van Krieken, E., and Thanapalasingam, T. (2022). Prompting as probing: Using language models for knowledge base construction. In Singhania, S., Nguyen, T.-P., and Razniewski, S., editors, *LM-KBC 2022 Knowledge Base Construction from Pre-trained Language Models 2022*, volume 3274 of *CEUR Workshop Proceedings*, pages 11–34. CEUR-WS.org. DOI: 10.48550/arxiv.2208.11057.
- Bentahar, J., Moulin, B., and Bélanger, M. (2010). A taxonomy of argumentation models used for knowledge representation. *Artificial Intelligence Review*, 33(3):211–259. DOI: 10.1007/s10462-010-9154-1.
- Bhatti, M. M. A., Ahmad, A. S., and Park, J. (2021). Argument mining on twitter: A case study on the planned parenthood debate. In Al-Khatib, K., Hou, Y., and Stede, M., editors, *Proceedings of the 8th Workshop on Argument Mining*, pages 1–11, Punta Cana, Dominican Republic. Association for Computational Linguistics. DOI: 10.18653/v1/2021.argmining-1.1.
- Boltužić, F. and Šnajder, J. (2015). Identifying prominent arguments in online debates using semantic textual similarity. In Cardie, C., editor, *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 110–115, Denver, CO. Association for Computational Linguistics. DOI: 10.3115/v1/W15-0514.
- Boltužić, F. and Šnajder, J. (2016). Fill the gap! analyzing implicit premises between claims from online debates. In Reed, C., editor, *Proceedings of the Third Workshop on Argument Mining (ArgMining2016)*, pages 124–133, Berlin, Germany. Association for Computational Linguistics. DOI: 10.18653/v1/W16-2815.
- Cabrio, E. and Villata, S. (2018). Five years of argument mining: a data-driven analysis. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 5427–5433. International Joint Conferences on Artificial Intelligence Organization. DOI: 10.24963/ijcai.2018/766.
- Carstens, L., Toni, F., and Evripidou, V. (2014). Argument mining and social debates. In *Comma*. DOI: 10.3233/978-1-61499-436-7-451.
- Chakrabarty, T., Hidey, C., Muresan, S., McKeown, K., and Hwang, A. (2019). Ampersand: Argument mining for persuasive online discussions. In Inui, K., Jiang, J., Ng, V., and Wan, X., editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2933–2943, Hong Kong, China. Association for Computational Linguistics. DOI: 10.18653/v1/D19-1291.
- Chen, G., Cheng, L., Luu, A. T., and Bing, L. (2024). Exploring the potential of large language models in computational argumentation. In Ku, L.-W., Martins, A., and Srikumar, V., editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2309–2330, Bangkok, Thailand. Association for Computational Linguistics. DOI: 10.18653/v1/2024.acl-long.126.
- Chen, W.-F., Chen, M.-H., Mudgal, G., and Wachsmuth, H. (2022). Analyzing culture-specific argument structures in learner essays. In Lapesa, G., Schneider, J., Jo, Y., and Saha, S., editors, *Proceedings of the 9th Workshop on Argument Mining*, pages 51–61, Online and in Gyeongju,

<sup>7</sup><https://github.com/AINDA-Project-UFCG>

- Republic of Korea. International Conference on Computational Linguistics. Available at: <https://aclanthology.org/2022.argmining-1.4/>.
- Duthie, R., Budzynska, K., and Reed, C. (2016). *Mining Ethos in Political Debate*, volume 287 of *Frontiers in Artificial Intelligence and Applications*, pages 299–310. IOS Press, Netherlands. This research was supported in part by EPSRC in the UK under grant EP/M506497/1 and in part by the Polish National Science Centre under grant 2015/18/M/HS1/00620.. DOI: 10.3233/978-1-61499-686-6-299.
- Habernal, I. and Gurevych, I. (2015). Exploiting debate portals for semi-supervised argumentation mining in user-generated web discourse. In Márquez, L., Callison-Burch, C., and Su, J., editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2127–2137, Lisbon, Portugal. Association for Computational Linguistics. DOI: 10.18653/v1/D15-1255.
- Habernal, I. and Gurevych, I. (2016). Which argument is more convincing? analyzing and predicting convincingness of web arguments using bidirectional lstm. In Erk, K. and Smith, N. A., editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1589–1599, Berlin, Germany. Association for Computational Linguistics. DOI: 10.18653/v1/P16-1150.
- Habernal, I. and Gurevych, I. (2017). Argumentation mining in user-generated web discourse. *Computational Linguistics*, 43(1):125–179. DOI: 10.1162/COLI<sub>a0</sub>0276.
- Haddadan, S., Cabrio, E., and Villata, S. (2019). Yes, we can! mining arguments in 50 years of us presidential campaign debates. In Korhonen, A., Traum, D., and Márquez, L., editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4684–4690, Florence, Italy. Association for Computational Linguistics. DOI: 10.18653/v1/P19-1463.
- Hautli-Janisz, A., Kikteva, Z., Siskou, W., Gorska, K., Becker, R., and Reed, C. (2022). Qt30: A corpus of argument and conflict in broadcast debate. In Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Odijk, J., and Piperidis, S., editors, *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3291–3300, Marseille, France. European Language Resources Association. Available at: <https://aclanthology.org/2022.lrec-1.352/>.
- Kotelnikov, E., Loukachevitch, N., Nikishina, I., and Panchenko, A. (2022). Ruarg-2022: Argument mining evaluation. In *Computational Linguistics and Intellectual Technologies*. RSUH. DOI: 10.28995/2075-7182-2022-21-333-348.
- Lavee, T., Orbach, M., Kotlerman, L., Kantor, Y., Gretz, S., Dankin, L., Jacovi, M., Bilu, Y., Aharonov, R., and Slonim, N. (2019). Towards effective rebuttal: Listening comprehension using corpus-wide claim mining. In Stein, B. and Wachsmuth, H., editors, *Proceedings of the 6th Workshop on Argument Mining*, pages 58–66, Florence, Italy. Association for Computational Linguistics. DOI: 10.18653/v1/W19-4507.
- Lawrence, J. and Reed, C. (2020). Argument mining: A survey. *Computational Linguistics*, 45(4):765–818. DOI: 10.1162/coli<sub>a0</sub>0364.
- Lima, P. L. and Campelo, C. E. (2024). Disfluency detection and removal in speech transcriptions via large language models. In *Anais do XV Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana*, pages 227–235, Porto Alegre, RS, Brasil. SBC. DOI: 10.5753/stil.2024.245417.
- Lippi, M. and Torroni, P. (2016). Argument mining from speech: Detecting claims in political debates. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1). DOI: 10.1609/aaai.v30i1.10384.
- Mancini, E., Ruggeri, F., Galassi, A., and Torroni, P. (2022). Multimodal argument mining: A case study in political debates. In Lapesa, G., Schneider, J., Jo, Y., and Saha, S., editors, *Proceedings of the 9th Workshop on Argument Mining*, pages 158–170, Online and in Gyeongju, Republic of Korea. International Conference on Computational Linguistics. Available at: <https://aclanthology.org/2022.argmining-1.15>.
- Marvin, G., Hellen, N., Jjingo, D., and Nakatumba-Nabende, J. (2024). Prompt engineering in large language models. In Jacob, I. J., Piramuthu, S., and Falkowski-Gilski, P., editors, *Data Intelligence and Cognitive Informatics*, pages 387–402, Singapore. Springer Nature Singapore. DOI: 10.1007/978-981-99-7962-2\_30.
- Mestre, R., Middleton, S. E., Ryan, M., Gheasi, M., Norman, T., and Zhu, J. (2023). Augmenting pre-trained language models with audio feature embedding for argumentation mining in political debates. In Vlachos, A. and Augenstein, I., editors, *Findings of the Association for Computational Linguistics: EACL 2023*, pages 274–288, Dubrovnik, Croatia. Association for Computational Linguistics. DOI: 10.18653/v1/2023.findings-eacl.21.
- Mestre, R., Milicin, R., Middleton, S. E., Ryan, M., Zhu, J., and Norman, T. J. (2021). M-arg: Multimodal argument mining dataset for political debates with audio and transcripts. In Al-Khatib, K., Hou, Y., and Stede, M., editors, *Proceedings of the 8th Workshop on Argument Mining*, pages 78–88, Punta Cana, Dominican Republic. Association for Computational Linguistics. DOI: 10.18653/v1/2021.argmining-1.8.
- Mirkin, S., Jacovi, M., Lavee, T., Kuo, H.-K., Thomas, S., Sager, L., Kotlerman, L., Venezian, E., and Slonim, N. (2018). A recorded debating dataset. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA). DOI: 10.48550/arxiv.1709.06438.
- Mirzakhmedova, N., Gohsen, M., Chang, C. H., and Stein, B. (2024). Are large language models reliable argument quality annotators? In Cimiano, P., Frank, A., Kohlhase, M., and Stein, B., editors, *Robust Argumentation Machines*, pages 129–146, Cham. Springer Nature Switzerland. DOI: 10.1007/978-3-031-63536-6\_8.
- Nguyen, H. (2018). *Context-aware Argument Mining and Its Applications in Education*. PhD thesis. Available at: <http://d-scholarship.pitt.edu/33316/>.

- Peldszus, A. (2014). Towards segment-based recognition of argumentation structure in short texts. In Green, N., Ashley, K., Litman, D., Reed, C., and Walker, V., editors, *Proceedings of the First Workshop on Argumentation Mining*, pages 88–97, Baltimore, Maryland. Association for Computational Linguistics. DOI: 10.3115/v1/W14-2112.
- Pojoni, M.-L., Dumani, L., and Schenkel, R. (2023). Argument-mining from podcasts using chatgpt. In Malburg, L. and Verma, D., editors, *Proceedings of the Workshops at the 31st International Conference on Case-Based Reasoning (ICCBR-WS 2023)*, volume 3438 of *CEUR Workshop Proceedings*, pages 129–144, Aberdeen, Scotland. CEUR. Available at: [https://ceur-ws.org/Vol-3438/paper\\_10.pdf](https://ceur-ws.org/Vol-3438/paper_10.pdf).
- Rajasekharan, A., Zeng, Y., and Gupta, G. (2023). argument analysis using answer set programming and semantics-guided large language models. In *ICLP'23 Workshop on Goal-directed Execution of Answer Set Programs*. Available at: <http://platon.etsii.urjc.es/~jarias/gde23/papers/06-Abhiramon.pdf>.
- Reed, C. and Norman, T., editors (2003). *Argumentation Machines: New Frontiers in Argument and Computation*. Argumentation Library. Kluwer Academic Publishers, Netherlands. Book.
- Rocha, G., Cardoso, H. L., Belouadi, J., and Eger, S. (2023). Cross-genre argument mining: Can language models automatically fill in missing discourse markers? *Argument Computation*, vol. Pre-press, no. Pre-press, pp. 1-41, 2024. DOI: 10.3233/aac-230008.
- Sazid, M. T. and Mercer, R. E. (2022). A unified representation and a decoupled deep learning architecture for argumentation mining of students' persuasive essays. In Lapesa, G., Schneider, J., Jo, Y., and Saha, S., editors, *Proceedings of the 9th Workshop on Argument Mining*, pages 74–83, Online and in Gyeongju, Republic of Korea. International Conference on Computational Linguistics. Available at: <https://aclanthology.org/2022.argmining-1.6>.
- Schneider, J. (2014). Automated argumentation mining to the rescue? envisioning argumentation and decision-making support for debates in open online collaboration communities. In Green, N., Ashley, K., Litman, D., Reed, C., and Walker, V., editors, *Proceedings of the First Workshop on Argumentation Mining*, pages 59–63, Baltimore, Maryland. Association for Computational Linguistics. DOI: 10.3115/v1/W14-2108.
- Sousa, J. P., Nascimento, R., Araujo, R., and Coelho, O. (2021). Não se perca no debate! mineração de argumentação em redes sociais. In *Anais do X Brazilian Workshop on Social Network Analysis and Mining*, pages 139–150, Porto Alegre, RS, Brasil. SBC. DOI: 10.5753/brasnam.2021.16132.
- Soyusiawaty, D. and Rahmawanto, F. (2018). Similarity detector on the student assignment document using levenshtein distance method. In *2018 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, pages 656–661. DOI: 10.1109/ISRITI.2018.8864339.
- Stylianou, N. and Vlahavas, I. (2021). Transformed: End-to-End transformers for evidence-based medicine and argument mining in medical literature. *Journal of Biomedical Informatics*, 117:103767. DOI: 10.1016/j.jbi.2021.103767.
- Van der Meer, M., Liscio, E., Jonker, C., Plaat, A., Vossen, P., and Murukannaiah, P. (2024). A hybrid intelligence method for argument mining. *Journal of Artificial Intelligence Research*, 80:1187–1222. DOI: 10.1613/jair.1.15135.
- van der Meer, M., Reuver, M., Khurana, U., Krause, L., and Baez Santamaria, S. (2022). Will it blend? mixing training paradigms & prompting for argument quality prediction. In Lapesa, G., Schneider, J., Jo, Y., and Saha, S., editors, *Proceedings of the 9th Workshop on Argument Mining*, pages 95–103, Online and in Gyeongju, Republic of Korea. International Conference on Computational Linguistics. Available at: <https://aclanthology.org/2022.argmining-1.8/>.
- van Eemeren, F. H., Garssen, B., Krabbe, E. C. W., Snoeck Henkemans, A. F., Verheij, B., and Wagemans, J. H. M. (2014). Handbook of argumentation theory. DOI: 10.1007/978-90-481-9473-5.
- Visser, J., Lawrence, J., Wagemans, J., and Reed, C. (2019). An annotated corpus of argument schemes in us election debates. In *Proceedings of the 9th Conference of the International Society for the Study of Argumentation (ISSA)*, 3-6 July 2018, pages 1101–1111. Available at: <https://arg.tech/people/chris/publications/2018/issa2018.pdf>.
- Walton, D., Reed, C., and Macagno, F. (2008). *Argumentation Schemes*. Cambridge University Press. DOI: 10.1017/cbo9780511802034.
- Westermann, H., Savelka, J., Walker, V. R., Ashley, K. D., and Benyekhlef, K. (2022). Toward an intelligent tutoring system for argument mining in legal texts. *IOS Press*. Available at: <https://ebooks.iospress.nl/pdf/doi/10.3233/FAIA220456>.

## A Single-Prompt Strategy

### A.1 Prompt for Argument and Its Components

---

```
system = "You are a proficient debate
analyst capable of identifying and
analyzing argumentative aspects in debate
transcriptions."
```

```
prompt = f"""
Given an audio transcription text from a
debate, delimited by triple backticks,
about 'Generative AI and its impact on
society,' your objective is to analyze the
argumentative aspects of the text following
the protocol described below. The response
must be a valid JSON file without any prose.
```

```
Follow these steps to analyze the
transcription debates:
```

```
1 - Identify the text spans of arguments.
Some speeches might not have an argument,
while others might have more than one. Using
the following definition below:
```

```
- Argument: An argument is a span of
text, within a speech or dialogue, that
contains a discursive construction composed
of argumentative elements, such as premises
and evidence, which are presented in a
logical and structured manner to support or
refute a claim. The purpose of the argument
is to persuade the listener or reader about
the validity or invalidity of the claim
in question, providing a coherent set of
reasoning and evidence that justifies the
defended position. A single utterance can
contain none or more than one argument.
Considering that it is not mandatory for
an argument to contain these components, it
must, at a minimum, be composed of claims
logically connected to evidence or premises.
```

```
2 - If you do not find any argument in the
text, return a JSON in the format with blank
strings as in the example below:
```

```
{
 "argument": [],
 "claim": [],
 "premise": [],
 "evidence": []
}
```

```
3 - If you find arguments, for each argument
get the text from the argument and look
for the argument's components using the
definitions below. Every argument must have a
claim and at least one evidence or premise.
```

```
- Evidence: It is a component of an
argument where we can find information or
data that supports or contests a claim.
These are essential to back up claims or
validate propositions. They may include
facts, statistics, expert opinions, concrete
examples, or historical events, offering
logical and substantial support to a claim.
The text span of evidence is contained within
the text span of an argument, and often,
evidence is associated with a premise.
```

```
- Claim: A claim is a component of an
argument in which a statement or proposition
is presented, expressing a position, opinion,
or declaration that can be subject to
evaluation. Claims are fundamental to
constructing arguments and supporting
positions and may or may not be backed
by evidence, logical reasoning, and other
persuasive elements. The text span of a claim
may or may not be contained within the text
span of an argument. It is possible for a
debater to make a claim without it being part
of an argument (an opinion without evidence).
```

```
- Premise: A certainty, belief,
statement, or proposition about the subject
matter that is used to develop an argument
supporting or opposing a claim. In other
words, it is a proposition, assumed or
accepted as true, that underpins and
justifies a conclusion or point of view.
```

```
4 - Do not rewrite what's in the original
text; it must be the same text from the
input in the output. (Copy and paste the
span of text that represents an argument or
argumentative component)
```

```
5 - Place the results in the output as
described below in the output format. The
output format must be a JSON file with the
following structure. Only generate valid JSON
and no prose:
```

```
{
 "argument": ["span of text with
argument 1", "span of text with argument 2"],
 "claim": ["span of text with claim
1", "span of text with claim 2"],
 "premise": ["span of text with
premise 1", "span of text with premise 2"],
 "evidence": ["span of text with
evidence 1", "span of text with evidence 2"]
}
```

```
6 - Follow the examples input and output
below:
```

```
{examples_string}

``` {text} ```

"""
```

B Two-Prompts Strategy

B.1 Argument prompt

```
system = "You are a proficient debate analyst  
capable of identifying arguments in debate  
transcriptions."
```

```
prompt = f"""
Given an argument from an audio transcription
text from a debate, delimited by triple
backticks, about 'Generative AI and its
impact on society', your objective is to
identify the arguments as described below.
The response must be a valid JSON file
without any prose.
```

Follow the steps to analyze the transcription debates:

1 - Identify the text spans of arguments. Some speeches might not have an argument, while others might have more than one.

- Argument: An argument is a 'span' of text, within a speech of a dialogue, that contains a discursive construction composed of argumentative elements, such as premises and evidence, which are presented in a logical and structured manner to support or refute a claim. The purpose of the argument is to persuade the listener or reader about the validity or invalidity of the claim in question, providing a coherent set of reasoning and evidence that justify the defended position. A single utterance can contain none more than one argument. Considering that it is not mandatory for an argument to contain all these structures, it must, at a minimum, be composed of claims logically connected to evidence or premises.

2 - If you do not find any argument in the text, return a JSON in the format with an empty list as in the example below:

```
{
  'argument': []
}
```

3 - Do not rewrite what's in the original text; it must be the same text from the

input in the output. (Copy and paste the span of text that represents an argument or argumentative component)

4 - Follow the examples input and output below:

```
{examples_string_only_argument}

``` {text} ```

"""
```

---

### B.2 Components Prompt

---

```
system = "You are a proficient debate
analyst capable of identifying and
analyzing argumentative aspects in debate
transcriptions."
```

```
prompt = f"""
Given an audio transcription text from a
debate, delimited by triple backticks, about
'Generative AI and its impact on society,'
your objective is to identify the argumentative
components from the text following the
protocol described below. The response must
be a valid JSON file without any prose.
```

Follow the steps to analyze the arguments:

1 - Identify the text spans of the components as described below:

- Evidence: It is a component of an argument where we can find information or data that supports or contests a claim. These are essential to back up claims or validate propositions. They may include facts, statistics, expert opinions, concrete examples, or historical events, offering logical and substantial support to a claim.

The text span of evidence is contained within the text span of an argument, and often, evidence is associated with a premise.

- Claim: A claim is a component of an argument in which a statement or proposition is presented, expressing a position, opinion, or declaration that can be subject to evaluation. Claims are fundamental to constructing arguments and supporting positions and may or may not be backed by evidence, logical reasoning, and other persuasive elements. The text span of a claim may or may not be contained within the text span of an argument. It is possible for a debater to make a claim without it being part of an argument (an opinion without evidence).

- **Premisse:** A certainty, belief, statement, or proposition about the subject matter that is used to develop an argument supporting or opposing a claim. In other words, it is a proposition, assumed or accepted as true, that underpins and justifies a conclusion or point of view.

2 - If you do not find some of the components in the argument, return a JSON in the format with an empty list as in the example below:

```
{
 'claim': [],
 'premise': [],
 'evidence': []
}
```

3 - Do not rewrite what's in the original text; it must be the same text from the input in the output. (Copy and paste the span of text that represents an argument or argumentative component)

4 - Follow the examples input and output below:

```
{examples_string_only_components}

``` {text} ```

"""
```

- **Premise:** A certainty, belief, statement, or proposition about the subject matter that is used to develop an argument supporting or opposing a claim. In other words, it is

a proposition, assumed or accepted as true, that underpins and justifies a conclusion or point of view.

2 - If you do not find any premise in the text, return a JSON in the format with an empty list as in the example below:

```
{
  'premise': []
}
```

3 - Do not rewrite what's in the original text; it must be the same text from the input in the output. (Copy and paste the span of text that represents an argument or argumentative component)

4 - Follow the examples input and output below:

```
{examples_string_only_premise}

``` {text} ```

"""
```

---

## C Four-Prompts Strategy

### C.1 Premise Prompt

---

```
system = "You are a proficient debate analyst capable of identifying and analyzing argumentative aspects in debate transcriptions."
```

```
prompt = f"""
Given an argument from an audio transcription text from a debate, delimited by triple backticks, about 'Generative AI and its impact on society', your objective is to identify the
premises in the arguments as described below. The response must be a valid JSON file without any prose.
```

Follow the steps to analyze the transcription debates:

1 - Identify the text spans of premises. Some speeches might not have a premise, while others might have more than one.

### C.2 Claim Prompt

---

```
system = "You are a proficient debate analyst capable of identifying and analyzing argumentative aspects in debate transcriptions."
```

```
prompt = f"""
Given an argument from an audio transcription text from a debate, delimited by triple backticks, about 'Generative AI and its impact on society', your objective is to identify the claims in the arguments as described below. The response must be a valid JSON file without any prose.
```

Follow the steps to analyze the transcription debates:

1 - Identify the text spans of claims. Some speeches might not have a claim, while others might have more than one.

- **Claim:** A claim is a component of an argument in which a statement or proposition



is presented, expressing a position, opinion, or declaration that can be subject to evaluation.

Claims are fundamental to constructing arguments and supporting positions and may or may not be backed by evidence, logical reasoning, and other persuasive elements. The text span of a

claim may or may not be contained within the text span of an argument. It is possible for a debater to make a claim without it being part of an argument (an opinion without evidence).

2 - If you do not find any premise in the text, return a JSON in the format with an empty list as in the example below:

```
{
 'claim': []
}
```

3. Do not rewrite what's in the original text; it must be the same text from the input in the output. (Copy and paste the span of text that represents an argument or argumentative component)

4. Follow the examples input and output below:

```
{examples_string_only_claim}

''' {text} '''
"""
```

---

while others might have more than one.

- Evidence: It is a component of an argument where we can find information or data that supports or contests a claim. These are essential to back up claims or validate propositions.

They may include facts, statistics, expert opinions, concrete examples, or historical events, offering logical and substantial support to a claim.

The text span of evidence is contained within the text span of an argument, and often, evidence is associated with a premise.

2 - If you do not find any evidence in the text, return a JSON in the format with an empty list as in the example below:

```
{
 'evidence': []
}
```

3 - Do not rewrite what's in the original text; it must be the same text from the input in the output. (Copy and paste the span of text that represents an argument or argumentative component)

4 - Follow the examples input and output below:

```
{examples_string_only_evidence}

''' {text} '''
"""
```

---

### C.3 Evidence Prompt

---

system = "You are a proficient debate analyst capable of identifying and analyzing argumentative aspects in debate transcriptions."

prompt = f"""  
Given an argument from an audio transcription text from a debate, delimited by triple backticks, about 'Generative AI and its impact on society', your objective is to identify the evidences in the arguments as described below. The response must be a valid JSON file without any prose.

Follow the steps to analyze the transcription debates:

1 - Identify the text spans of evidences. Some speeches might not have a evidence,