Finanziato
dall'Unione europea
NextGenerationEU

Ministero
dell'Università
e della Ricerca

Italiadomani
PIANO NAZIONALE
DI RIPRESA E RESILIENZA

PRIN 2022 Project EPICA

Deliverable 3.3
Technical report and scientific publications describing the analysis of PIC case studies

Pietro Baroni[1], Stefano Bistarelli[2], Bettina Fazzinga[3], Giulio Fellin[1], Sergio Flesca[4], Filippo Furfaro[4], Massimiliano Giacomin[1], Francesco Parisi[4], Carlo Proietti[5], Irene Russo[5], Francesco Santini[2], Carlo Taticchi[2], and Paola Vernillo[5]

[1]DII - Universitá di Brescia

[2]DMI - Universitá di Perugia

[3]DICES - Universitá della Calabria

[4]DIMES - Universitá della Calabria

[5]ILC - Consiglio Nazionale delle Ricerche

Abstract
This document presents the main results of the WorkPackage 3 of the PRIN 2022 project EPICA. In particular, it introduces ArguGraph, an argument map induction pipeline that combines claim extraction, co-reference resolution, and textual entailment NLP modules powered by LLMs with graph-based heuristics.

1 Introduction

This document is the Deliverable 3.3 *Technical report and scientific publications describing the analysis of PIC case studies*[1] of the PRIN 2022 project

[1] This deliverable is subject to changes in the event that errors or omissions are detected after its release.

Empowering Public Interest Communication with Argumentation (EPICA), funded by European Union – Next Generation EU - Mission 4 “Education and Research” component C2 - investment 1.1 (CUP D53D23008860006).

This deliverable is the main outcome of the Work Package 3 “Model-based case analysis and model validation” of the EPICA project, and reports on the research activities we have undertaken to achieve theoretical and methodological advances in CA.

The document is organized as follows. Section 2 presents the definition of *Argument Mapping* and discusses previous attempts to partially implement argument mining pipelines. While Section 3 proposes to reframe the extraction of structurally organized argumentative content from a text as argument map induction rather than argument mining, Section 4 gives an overview of the ArguGraph algorithm. The dataset, experimental settings and results are presented in Section 5. Section 6 presents some preliminary conclusions and possible future perspectives for our work.

Each text or discourse can be argumentative to a certain extent. Automatically extracting what the writer/speaker believes and why s/he believes it is the aim of argument mining. Automatically extracting arguments has multiple application scenarios: legal document analysis [15, 46], fact-checking and misinformation detection [14], automatic analysis of political discourse [17], automatic evaluation of students’ writing skills [24], analysis of content from online debate and e-partecipation platforms [5]. Argument mining has the potential to transform fields that require reasoning, persuasion, or decision-making by enabling automated and structured analysis of arguments.

The vast majority of NLP approaches structure argument mining as a three-step process [20, 17]: argument identification, aiming at recognizing and extracting arguments from natural language; argument classification, aiming at classifying arguments as argument’s components (e.g., claims and premises) and, finally, argumentative relations prediction, aiming at labeling - in general, as support or attack - the relations between arguments previously identified.

However, this well-thought-out modular approach to the task has not made it solvable. Argument mining is a challenging task because i) manually annotated datasets that are essential for training and testing models are heterogeneous for the textual genre and annotation schemes, making the results less generalizable; ii) the pipeline is modular and, consequently, errors made at one level will propagate to the other levels, and iii) currently, argument mining pipelines are heavily anchored to the textual level and are unable to extract and connect non-contiguous argumentative elements [2], making hard the extraction and comparison of argumentative units from contexts that are not monological.

With the emergence of Large Language Models (LLMs) the challenge still persists, because LLMs are capable of performing well on a single-stage task based on knowledge extraction, but not on the derivation of structured content from text reconstructed and filtered through the reasoning capabilities of the model itself. Specific models trained for specific tasks have been developed [1, 32] but they need fine-tuning for the specific datasets analyzed and are not usable in a zero-shot scenario.

To handle these shortcomings, we propose a shift in perspective on this task introducing ArguGraph, an argument map induction pipeline that passes the outputs of claim extraction and co-reference resolution modules to LLMs devoted to NLI-based zero-shot classification. A set of graph-oriented heuristics enables the induction of an argument map as a directed graph [47]. Inspired by the literature on argument mapping as the process of structurally representing the relationships between arguments, claims, and evidence in a text through manual argument mapping tools like Araucaria [31] we aim at deriving from short argumentative texts their argumentative core. This work aims to propose an alternative methodology to argument mining for extracting the argumentative content of a text and to experimentally investigate, as a first step, the extent to which the emerging structure overlaps with that found in a text manually annotated for arguments.

As illustrated in Figure 1, ArguGraph outputs a graph-based representation of the argumentative content of a text by splitting it into sentences, summarizing the claims in each sentence, and selecting the most probable textual entailment relation for each pair of sentences. A set of theoretically motivated heuristics is then applied to derive a more concise subgraph from the graph-based representation through iterative pruning.

2 Related Work

2.1 Argument Mapping

Argument maps (also called argument diagrams) are tools that help users to visually and hierarchically depict the structure of an argument by making explicit the relationships between their constituents, e.g., premises and claims [30, 44, 23]. These two-dimensional graphs typically use colored boxes to represent propositions (e.g., the wording of the claim) as well as arrows and lines to highlight the inferential relationships between propositions (e.g., an associ-

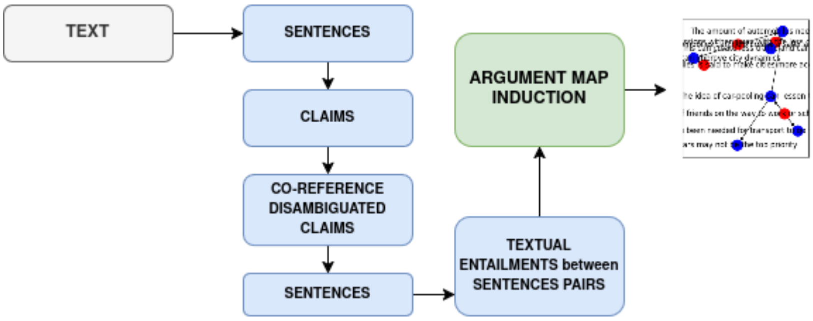
TEXT
SENTENCES
CLAIMS
CO-REFERENCE DISAMBIGUATED CLAIMS
SENTENCES
TEXTUAL ENTAILMENTS between SENTENCES PAIRS
ARGUMENT MAP INDUCTION

Figure 1: Overview of ArguGraph: the process begins with the extraction of claims from sentences splitted text (step 1). Extracted claims are disambiguated through coreference resolution and further splitted in sentences (step 2). Most probable textual entailment relations are assigned to each sentence pairs (step 3). Finally, thanks to heuristics a graph-based representation of the argumentative content is derived as an argument map (step 4).

ation of a claim with evidence). Argument maps have been used extensively in several different research fields, ranging from pedagogical to legal, and design domains [11, 43]. However, their popularity has recently soared in teaching and learning environments where these tools are deemed to be helpful for critical thinking and writing development [40], although experimental findings are not always consistent [30]. To date, quite a few software have been made available on the market to support users in the representation of argumentation knowledge [34]: among the others, ArgMap, Araucaria [31], Athena [22], Belvedere [38], Convince me [33], Digalo [35], LARGO [29], Rationale [43], and Reason!Able [42]. Unfortunately, there are no software tools capable of automatically inducing argument maps from texts, nor are there datasets available with manually generated argument maps aligned with the texts from which they were extracted. For this reason, in the present work, we evaluate the content of the induced argument maps with manual annotation performed for argument mining on a set of argumentative essays (Section 5.2).

2.2 Argument Mining Pipelines

The first attempt to partially implement an argument mining pipeline was MARGOT, a web-based system for argument mining designed for non-specialist users [20] [2]. Trained on the IBM corpus, it employs Tree Kernels for feature-rich analysis. MARGOT's versatility is tested for various genres, including

[2] http://margot.disi.unibo.it

Wikipedia, news articles, and Reddit comments. This system extracts argumentative sentences and identifies claim and evidence boundaries, but does not classify relations between components. Labeling of argument relations is also the missing step in the argument mining framework proposed by [17] which is structured around three key tasks leveraging fine-tuned large language models (argument detection, topic extraction, and argument stance classification). [24] propose a complete pipeline for automated essay scoring. By incorporating argumentative features such as claims, premises, and support relations into models devoted to this task, the performance on holistic scoring improved. Unfortunately, an implemented version of the pipeline is not available for testing. [2] shared the official implementation [3] of the sequential pipeline for full-text Scholarly Argumentation Mining (SAM) presented in their paper. The authors leverage SciBERT-based models to advance performance in argumentative discourse unit recognition (ADUR) and argumentative relation extraction (ARE). However, the pipeline is designed for a specific textual genre and its performance for different genres have not been assessed.

Previous approaches in argument mining have reformulated the relationships between arguments in terms of textual entailment, but only on previously identified argumentative units. [4] propose a framework to analyze online debates on Debatepedia, automatically labeling previously annotated argument relations with a TE open-source software based on the assumption that the probability of an entailment relation is inversely proportional to the editing distance. Their work is relevant because it represents the first attempt at using TE in argument mining. In their experiments, 0.67 is the accuracy obtained on a test set composed of 110 arguments. [16] use textual entailment for the Fact Extraction and VERification (FEVER) 2018 SemEval shared task to understand if factual sentences support or refute a factual claim. They train an ESIM (Enhanced Sequential Inference Model) originally developed for the SNLI dataset; they improve the organizers' baseline for the recognition of textual entailment. However, the FEVER dataset contains exclusively factual information while argumentation can be based also on not-factual premises and hypotheses. [7] present and evaluate ten datasets with different deep learning models containing annotated argumentative relations. They propose a set of strong cross-dataset baselines based on four neural architectures, with homogenous results over all datasets. They compare the performances obtained with non-contextualized and contextualised (e.g. BERT) word embeddings. Apart from embeddings, they also used sentiment and syntactic features plus a feature of entailment using AllenNLP, a TE model based on a decomposable attention model [28]. [32] apply five transformer-based models to predict argument rela-

[3] `https://github.com/DFKI-NLP/sam`

tions on five different domains, fine-tuning the models with data from US2016 debate corpus, the largest existing argument annotated corpus (more than 12,000 relations). The best model is RoBERTa-large, and results on different domains are comparable, showing that the model fine-tuned on US2016 is able to generalize on different data. Concerning previous work, our investigation is different because 1) we assume that the role of argumentative units can emerge from the overall network of relationships between the atomic units [21] of a text and 2) we use models that have not been fine-tuned for argumentative relations classification (contrary to [7] and [32]).

3 Reframing Argument Components Extraction

In this paper, we reframe the extraction of structurally organized argumentative content from a text as argument map induction rather than argument mining. This approach leverages textual entailment relationships between sentences. We assume that at least some of the entailment and contradiction relations between sentence pairs correspond to the inferential relations relevant to argument mapping and argument mining. For example, the support relation in b can be retrieved as a case of textual entailment (a):

(1) a. T: *If you help the needy, God will reward you.*
H: *Giving to the poor has good consequences.*

b. Premise: *Arts can enrich people's cultural lives.*
Claim: *Arts should by no means been disregarded by the government*

Argument mapping and argument mining are distinct tasks with different objectives. Argument maps abstract content from the text by summarizing it without prior linguistic segmentation [11, 12]. In terms of relationships, argument mapping focuses on logical rather than chronological or rhetorical structures [26]. However, argument mapping has yet to be validated by solid theories of informal logic [30]. In contrast, textual entailment allows for a more flexible directional relationship between text fragments than strict logical entailment [9].

The quality of an argument map is evaluated based on structural criteria linked to the enhancement of critical thinking—one of the main pedagogical uses of argument maps. For instance, generating and rebutting counterarguments is a cognitively demanding skill [18]. Structurally complex arguments that include all argument components are generally considered superior. The complexity of an argument is strongly influenced by the presence or absence of rebuttals [13]. Additionally, the density of supporting evidence is a crucial factor: arguments that provide relevant data and information are perceived as stronger [6]. The inclusion of evidence is fundamental in constructing a solid argument [10, 25, 45, 41].

These criteria highlight that argument maps can structure content differently from manual argumentative annotations. For instance, argument mapping can introduce relationships between arguments not explicitly connected by discourse markers and may be located far apart or may be far apart.

Despite these differences, the final outputs of argument mapping and argument mining are comparable. Argument map components are strongly influenced by Toulmin's theory [39], which also informs several argument mining annotation schemes, including the one used for the Persuade 2.0 dataset [8] (see Section 5.1).

The key question is: To what extent does the structural organization of argumentatively annotated content for argument mining align with the structure of an argument map? This is a theoretical issue that could be addressed if a dataset of argumentatively annotated essays were available, following an argument mining scheme and supplemented with purposefully created argument maps. In the absence of such a dataset, this study adopts an experimental approach: Given the basic structure of the essays, how frequently do elements labeled with analogous tags through an iterative graph-pruning algorithm correspond to those labeled in an argument mining scheme?

4 ArguGraph

We provide an overview of the ArguGraph algorithm. The process begins by segmenting the text into sentences and extracting claims as atomic facts using a pre-trained claim extraction model [36] (Section 4.1). For each set of atomic facts, co-reference resolution is performed before the extracted content is again split into disambiguated sentences. Next, all possible permuted pairs of the split sentences are generated, and their most probable textual entailment relation is determined (Section 4.2). The argument map for each text is represented as a directed graph and the classification of sentences into argument components is made possible through iterative pruning of the graph thanks to the structural positions of the elements within the graph (Section 4.3).

4.1 Claim extraction

Given a text *t*, we split it into the set of sentences $S = \{s_1, s_2, \ldots, s_n\}$. Each sentence is processed by a claim extractor module that derives elementary information units from the text. The result is a set of textual units $T = \{t_1, t_2, \ldots, t_n\}$ such that $|S| = |T|$. Each unit in *T* corresponds to a summarised claim that can be composed of more than one sentence. The elements in set *T* require further processing as they are potentially ambiguous in terms of references. For this reason, every element within the set *T* is processed with a co-reference resolution model. Elements are further divided at the sentence level because extracted and disambiguated claims can be composed of more than one sentence. The output of this step is the set of sentences $S' = \{s_1, s_2, \ldots, s_n\}$ such that $|S'| > |S|$. However, we expect a partial isomorphism between the two sets, which will be verified during the evaluation phase (Section 5.2).

4.2 Textual entailment between sentence pairs

Given a set of elements representing all permuted sentence pairs,

$$P = \{(s_i, s_j) \mid s_i, s_j \in S' ands_i \neq s_j\} \tag{1}$$

and a set of properties,

$$T = \{\text{entailment, contradiction, neutral}\} \tag{2}$$

each sentence pair is assigned the most probable textual entailment relation using a function:

$$f : P \mapsto T \tag{3}$$

The result of this function is the output of a large language model (LLM) trained on natural language inference datasets performing zero-shot classification (Section 5.1).

4.3 Argument Map Induction

The output of the function f serves as the input for the argument map induction algorithm. By establishing arcs between nodes representing sentences in an entailment relationship, for each text *t* we create G, a directed graph with potentially multiple disconnected components. We identify as the argumentative core of the text the connected subgraph V^{*} such that

$$V^* = \arg\max_{V' \subseteq V} |V'| \tag{4}$$

where $V' \subseteq V$ and $|V'|$ is the subgraph's [4] The structural positions of the elements within the subgraph determine the class of the argument components:

- the sink node C of V^{*} is the position or major claim;
- if there is no sink node, the node with the maximum positive difference between its in-degree and out-degree nodes is identified as the position;
- the set L of in-degree nodes of node C are claims;

The subgraph V^{*} is selectively incremented with sentences in a contradiction relationship:

- for each node in L the set of in-degree nodes K are counterclaim;
- for each node in K the set of in-degree nodes R are rebuttals.

Through iterative pruning of nodes, the result is a directed graph with edges of two types (entailments and contradictions) that represent the argumentative core of the text processed.

A caveat: There is a fourth type of argumentative component, evidence, which Crossley defines as an idea or example that supports other argumentative elements. From a structural perspective, the methodology presented does not allow us to distinguish between a node that serves as evidence for the position and a node that is a claim without supporting evidence. For this reason, we have chosen to limit our analysis to the identification of claims, thereby overestimating their number.

5 Experiments and Results

5.1 NLP argumentative content processing

Dataset. There are multiple manually annotated datasets based on annotation schemes designed to identify the argumentative structure of a short text.

[4] However, smaller clusters of interconnected nodes remain relevant, as they may still carry significant argumentative value.

The experiments reported in this paper involve a subset of PERSUADE 2.0 [8]. PERSUADE 2.0 is a large-scale dataset consisting of over 25,000 argumentative essays written by 6th-12th grade students in the U.S. annotated essays using seven discourse types[5]. Leaving out the rhetorical types (e.g., lead and concluding statements), we focus on four argumentive components:

- **position**: an opinion or conclusion on the main topic, also called major claim in [37];
- **claim**: a claim that supports the position;
- **counterclaim**: a claim that refutes another claim;
- **rebuttal**: a claim that refutes a counterclaim.

We evaluate argument maps emerging from 194 essays about a specific topic, i.e. car-free cities that display the maximum complexity in terms of argumentative structure, i.e. they have at least one (and maximum 2) counterclaim-rebuttals pairs.

At the current stage, we will not evaluate evidence-type elements defined as ideas or examples that support the other argumentative components, as they cannot be unambiguously identified through the argument map induction algorithm.

Experimental setting and results. As described in Section 4.1, for each essay in the dataset we split the sentences using the sentence tokenizer module provided by nltk [3]. Using a distilled version of the claim extractor proposed by [36], each sentence is processed to collect all the potential claims. The claims have been disambiguated at the co-reference level using fastcoref [27]. The disambiguated textual content has been split into sentences and all the permuted pairs (except the pair consisting of the sentence with itself) are generated. For 194 essays we get 598,582 sentence pairs, with an average of 5471 pairs per essay (std = 4071). Each pair is annotated with three NLI-based zero-shot classification models[6]: bart-large-mnli [7], deberta-large-mnli [8], and mDeBERTa-v3-base-mnli-xnli [9]. As we can see from Table 1, the models differ in terms of the type of relationship identified as a percentage of the total. In particular, mDeBERTa-v3-base-mnli-xnli identifies significantly more contradictions compared to the other models. The unbalance persists after the induction of argument maps (see Table 2). Overestimating a relationship that

[5] The PERSUADE 2.0 dataset is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0) license.

[6] All the models are licensed under the MIT License.

[7] `https://huggingface.co/facebook/bart-large-mnli`

[8] `https://huggingface.co/microsoft/deberta-large-mnli`

[9] `https://huggingface.co/MoritzLaurer/mDeBERTa-v3-base-mnli-xnli`

model	ent	contr	neu
mDeBERTa	3.5%	11.9%	84.8%
deberta-large-mnli	26%	2.9%	71.2%
bart-large-mnli	18.4%	2.3%	79.6%

Table 1: Percentages of TE classes for three NLI classification models.

model	P	C	CC	R
mDeBERTa	1	9.08 ± 16.78	15.12 ± 16.62	26.43 ± 21.85
deberta	1	42.97 ± 43.29	6.92 ± 10.52	4.44 ± 7
bart	1	30.43 ± 28.12	8.59 ± 9.65	6.92 ± 9.93

Table 2: Average number and std of argumentive elements retrieved for each essay after iterative pruning of argument maps (P= position, C = claim, E = evidence, CC = counterclaim, R = rebuttal).

is known to be less frequent in the analyzed texts will result in a lower overall performance of this model.

num_nodes	P	C	CC	R
mDeBERTa	-0.29**	0.36**	0.12	-0.03
deberta	-0.39**	0.53***	0.26*	-0.01
bart	-0.25*	0.40***	0.28**	0.41***

Table 3: Pearson correlations between the number of nodes for each essay and the ROUGE F1 score of argumentative elements for each model.

5.2 Argument Maps Evaluation

Alternative argument maps emerge from different zero-shot NLI models after the induction of the argument map. We evaluate the results measuring with ROUGE metric [19] if the elements identified as positions, claims, counterclaims, and rebuttals are equal or part of the rispective textual elements manually annotated in Persuade 2.0.

For the argumentative type Position, we evaluate the models by computing ROUGE-1 F1 between the annotated position in each argumentative essay and the corresponding node identified through argument map induction.

Interestingly, there is an overlap among the set of identified position nodes. This indicates that, regardless of the differing numbers of entailments and contradictions identified by each model, the methodology described in Section 4.3 allows for the unique abstraction of the major claim from the network of relationships in an argumentation. However, as shown in Figure 2, the position is definitively identified in only a minority of the essays.

We recall that no content filtering was applied to the analyzed essays, which certainly contain rhetorical textual elements that should not be included in an argumentative structure. Moreover, the failure to identify certain elements could be due to the variable performance of the argument map induction pipeline on essays that, once processed, appear noisier than others. For example, claims that are not explicitly textual or particularly short sentences are unlikely to be argumentative components. To assess whether the number of nodes in the graph affects the quality of the automatically identified argumentative components, Table 3 presents the correlations for each model and each argumentative type.

For all models, there is a significant inverse correlation between the number of nodes and the ROUGE-1 F1 score for elements classified as Position. This suggests that an excessive number of textual entailment relationships, leading to a higher probability of noise in the graph, negatively impacts the identification of this element.

Concerning claims, in a conservative approach, we considered only the node with the highest ROUGE-1 F1 score for each essay. We aimed to determine whether the pipeline was able to confidently identify at least one of the claims manually annotated in the Persuade 2.0 essays. As shown in Figure 3, mDeBERTa predictably achieves the worst results, as it is the model that underestimates entailment relationships. In this case, the correlation between the number of nodes and claim identification is positive and significant for all models: having more nodes increases the likelihood of identifying at least one argumentative element classified as a claim, despite the potential negative effects on Position identification.

Finally, concerning counterclaim and rebuttal elements, each of which appears only once per essay in the analyzed dataset, we selected the node with the highest ROUGE-1 F1 score among those identified as counterclaims and rebuttals, similar to what was done for claims. Since the identification of these nodes is based on the contradiction relation, it is easier for mDeBERTa (Figure 4) which overestimates the contradiction relations. However, for this model there is no significant correlation with the number of nodes.

In contrast, for BART, there is a positive and significant correlation, particularly

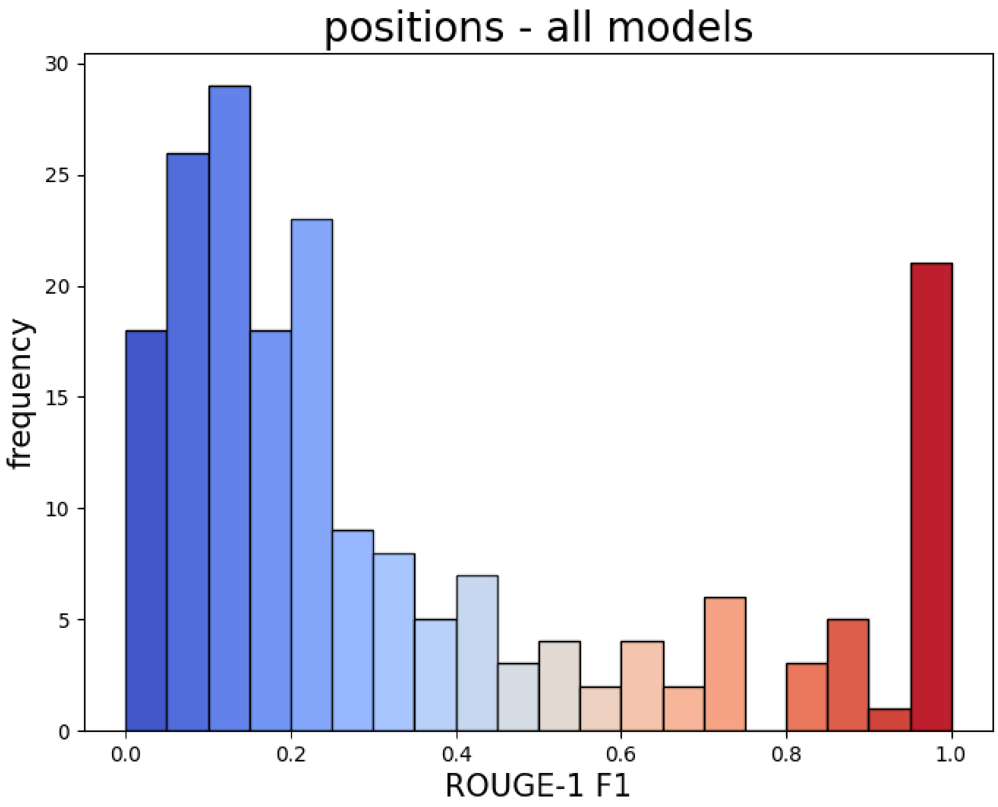

Figure 2: Distribution of ROUGE-1 F1 scores for position identification across all models. The histogram illustrates the frequency of different score ranges.

in the case of rebuttals. Overall, BART emerges as the most promising model, making it the primary candidate for future implementations aimed at pruning noisy nodes from the graph.

6 Conclusions and Future Work

Argument mining remains a challenging task, hindered by heterogeneous datasets, error propagation in modular pipelines, and limitations in handling non-contiguous argumentative elements. ArguGraph addresses these challenges by leveraging claim extraction, co-reference resolution, and textual entailment modules with graph-based heuristics, offering a novel approach for the induction of structured argumentative content as an argument map. Our experiments demonstrate that the induction of argument maps from texts partially identifies

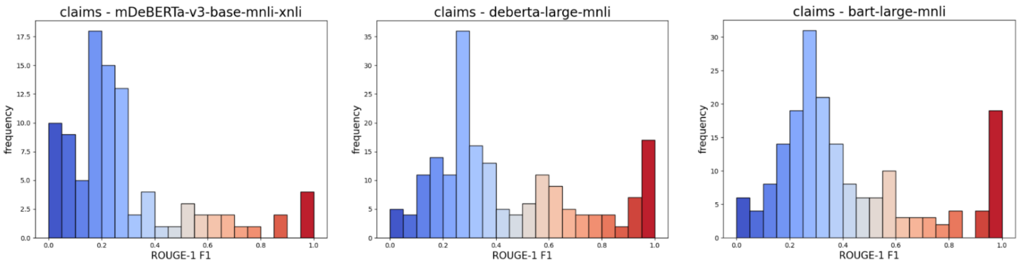

Figure 3: Comparison of ROUGE-1 F1 score distributions for claim identification across different models. Each histogram represents the frequency of scores for mDeBERTa-v3-base-mnli-xnli (left), deberta-large-mnli (center), and bart-large-mnli (right).

the manually annotated argumentative structure even in a zero-shot classification scenario of argumentative essays that contain no argumentative components. In future work, further heuristics for iterative pruning of the graph will be tested. We also plan to implement argument map-inspired metrics to measure the qualitative evaluation of the argument maps.

Limitations

Our approach - as with every argument mining approach - integrates different components (i.e., textual entailment, co-reference resolution, and claim extraction). The potential errors introduced may affect the overall quality of the induced argument map, and therefore future investigations with new models and/or error correction strategies are needed. However, our pipeline does not aim to extract the complete set of argumentative elements from a text but a summarized version of them, thus only the presence of significant structural gaps proves problematic.

Getting textual entailment relations for all sentence pairs is computationally expensive for longer text. This means that a methodology to identify the most argumentatively dense parts of a long text for automatic processing with the pipeline is necessary.

Although the pipeline is general enough to be implemented for any language, it is undeniable that the performance level of individual modules for less-supported languages compared to English impacts expectations regarding the output.

Finally, this paper does not present impressive results in terms of performance

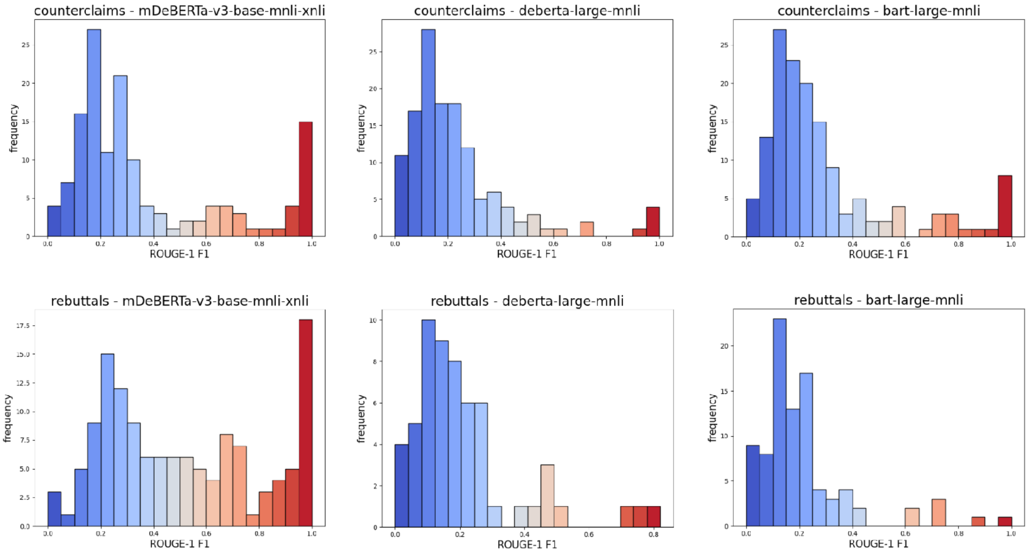

Figure 4: Comparison of ROUGE-1 F1 score distributions for counterclaim and rebuttal identification across different models. Each histogram represents the frequency of scores for mDeBERTa-v3-base-mnli-xnli (left), deberta-large-mnli (center), and bart-large-mnli (right).

on recognized benchmarks.We believe that understanding the argumentative structure of a text or discourse is a cognitively complex task that cannot be solved solely through large language models, even by improving their performance on individual tasks. On the contrary, it requires a rethinking of the implementation design. Our contribution, however modest and preliminary, is a step in that direction.

References

[1] Gregor Betz and Kyle Richardson. Deepa2: A modular framework for deep argument analysis with pretrained neural text2text language models. In *STARSEM*, 2021.

[2] Arne Binder, Bhuvanesh Verma, and Leonhard Hennig. Full-text argumentation mining on scientific publications. *ArXiv*, abs/2210.13084, 2022.

[3] Steven Bird. Nltk: The natural language toolkit. In *Annual Meeting of the Association for Computational Linguistics*, 2006.

[4] Elena Cabrio and Serena Villata. A natural language bipolar argumentation approach to support users in online debate interactions†. *Argument & Computation*, 4(3):209–230, 2013.

[5] Tuhin Chakrabarty, Christopher Hidey, Smaranda Muresan, Kathy McKeown, and Alyssa Hwang. AMPERSAND: Argument mining for PERSuAsive oNline discussions. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, 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, November 2019. Association for Computational Linguistics.

[6] Kyoo-Lak Cho and David H. Jonassen. The effects of argumentation scaffolds on argumentation and problem solving. *Educational Technology Research and Development*, 50:5–22, 2002.

[7] Oana Cocarascu, Elena Cabrio, Serena Villata, and Francesca Toni. A dataset independent set of baselines for relation prediction in argument mining, 2020.

[8] S.A. Crossley, Y. Tian, P. Baffour, A. Franklin, M. Benner, and U. Boser. A large-scale corpus for assessing written argumentation: Persuade 2.0. *Assessing Writing*, 61:100865, 2024.

[9] Ido Dagan, Oren Glickman, and Bernardo Magnini. The pascal recognising textual entailment challenge. In Joaquin Quiñonero-Candela, Ido Dagan, Bernardo Magnini, and Florence d'Alché Buc, editors, *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, pages 177–190, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[10] Darmawansah Darmawansah, Dzul Rachman, Febiyani Febiyani, and Gwo-Jen Hwang. Chatgpt-supported collaborative argumentation: Integrating collaboration script and argument mapping to enhance efl students' argumentation skills. *Education and Information Technologies*, 2024.

[11] Martin Davies. Computer-assisted argument mapping: a rationale approach. *Higher Education*, 58:799–820, 2009.

[12] Christopher P. Dwyer, Michael Hogan, and Ian Stewart. The evaluation of argument mapping as a learning tool: Comparing the effects of map

reading versus text reading on comprehension and recall of arguments. *Thinking Skills and Creativity*, 5:16–22, 2010.

[13] Sibel Erduran, Shirley Simon, and Jonathan F. Osborne. Tapping into argumentation: Developments in the application of toulmin's argument pattern for studying science discourse. *Science Education*, 88:915–933, 2004.

[14] Max Glockner, Yufang Hou, Preslav Nakov, and Iryna Gurevych. Missci: Reconstructing fallacies in misrepresented science. In *Annual Meeting of the Association for Computational Linguistics*, 2024.

[15] Ivan Habernal, Daniel R. Faber, Nicola Recchia, Sebastian Bretthauer, Iryna Gurevych, Indra Spiecker genannt Dohmann, and Chr. Burchard. Mining legal arguments in court decisions. *Artificial Intelligence and Law*, pages 1–38, 2022.

[16] Andreas Hanselowski, H. Zhang, Zile Li, Daniil Sorokin, Benjamin Schiller, Claudia Schulz, and Iryna Gurevych. Ukp-athene: Multi-sentence textual entailment for claim verification. *ArXiv*, abs/1809.01479, 2018.

[17] Arman Irani, Ju Yeon Park, Kevin E. Esterling, and Michalis Faloutsos. Wiba: What is being argued? a comprehensive approach to argument mining. *ArXiv*, abs/2405.00828, 2024.

[18] Deanna Kuhn. Thinking as argument. *Harvard Educational Review*, 62:155–178, 1992.

[19] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Annual Meeting of the Association for Computational Linguistics*, 2004.

[20] Marco Lippi and Paolo Torroni. Margot: A web server for argumentation mining. *Expert Systems with Applications*, 65:292–303, 2016.

[21] Qing Liu, Zhiying Zhong, and John C. Nesbit. Argument mapping as a pre-writing activity: Does it promote writing skills of efl learners? *Educ. Inf. Technol.*, 29:7895–7925, 2023.

[22] Charlotte Magnusson and Bertil Rolf. Developing the art of argumentation - a software approach. In *Proceedings of the 5th International Conference on Argumentation. International Society for the Study of Argumentation (ISSA-2002)*, 2002.

[23] Mike Metcalfe and Saras Sastrowardoyo. Complex project conceptualisation and argument mapping. *International Journal of Project Management*, 31:1129–1138, 2013.

[24] Huy V. Nguyen and Diane J. Litman. Improving argument mining in student essays by learning and exploiting argument indicators versus essay topics. In *The Florida AI Research Society*, 2016.

[25] Omid Noroozi, Seyyed Kazem Banihashem, Harm J. A. Biemans, Mattijs Smits, M.T.W. Vervoort, and Caro-Lynn Verbaan. Design, implementation, and evaluation of an online supported peer feedback module to enhance students' argumentative essay quality. *Education and Information Technologies*, pages 1 – 28, 2023.

[26] Alexandra Okada and Simon Buckingham Shum. Evidence-based dialogue maps as a research tool to investigate the quality of school pupils' scientific argumentation. *International Journal of Research & Method in Education*, 31:291 – 315, 2008.

[27] Shon Otmazgin, Arie Cattan, and Yoav Goldberg. F-coref: Fast, accurate and easy to use coreference resolution. In *AACL*, 2022.

[28] Ankur P. Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. A decomposable attention model for natural language inference. In *EMNLP*, 2016.

[29] Niels Pinkwart, Vincent Aleven, Kevin D. Ashley, and Collin Lynch. Toward legal argument instruction with graph grammars and collaborative filtering techniques. In *International Conference on Intelligent Tutoring Systems*, page 227–236, 2006.

[30] Chrysi Rapanta and Douglas Walton. The use of argument maps as an assessment tool in higher education. *International Journal of Educational Research*, 79:211–221, 2016.

[31] Chris Reed and Glenn Rowe. Araucaria: Software for argument analysis, diagramming and representation. *Int. J. Artif. Intell. Tools*, 13:983–, 2004.

[32] Ramon Ruiz-Dolz, Jose Alemany, Stella Heras, and Ana Garcia-Fornes. Transformer-based models for automatic identification of argument relations: A cross-domain evaluation. *IEEE Intelligent Systems*, page 1–1, 2021.

[33] Patricia K. Schank and Michael Andrew Ranney. Improved reasoning with convince me. In R.L. Mack J. Miller, I.R. Katz and L. Marks, editors, *Human Factors in Computing Systems, CHI '95 Conference Companion: Mosaic of Creativity, Denver, Colorado, USA, May 7-11, 1995*, pages 276–277. ACM, 1995.

[34] Oliver Scheuer, Frank Loll, Niels Pinkwart, and Bruce M. McLaren. Computer-supported argumentation: A review of the state of the art. *International Journal of Computer-Supported Collaborative Learning*, 5:43–102, 2010.

[35] Baruch B. Schwarz and Amnon Glassner. The role of floor control and of ontology in argumentative activities with discussion-based tools. *International Journal of Computer-Supported Collaborative Learning*, 2:449–478, 2005.

[36] Alessandro Scirè, Karim Ghonim, and Roberto Navigli. FENICE: Factuality evaluation of summarization based on natural language inference and claim extraction. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14148–14161, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[37] Christian Stab and Iryna Gurevych. Parsing argumentation structures in persuasive essays. *Computational Linguistics*, 43(3):619–659, September 2017.

[38] Daniel D. Suthers, Arlene Weiner, John Connelly, and Massimo Paolucci. Belvedere: Engaging students in critical discussion of science and public policy issues. In J. Greer, editor, *Proceedings of the 7th World Conference on Artificial Intelligence in Education (AI-ED 1995)*, page 266–273, 1995.

[39] Stephen E. Toulmin. The uses of argument. 1960.

[40] Charles R. Twardy. Argument maps improve critical thinking. *Teaching Philosophy*, 27:95–116, 2004.

[41] Susan W. van den Braak, Herre van Oostendorp, Henry Prakken, and Gerard A. W. Vreeswijk. A critical review of argument visualization tools: Do users become better reasoners? 2006.

[42] Timothy van Gelder. Argument mapping with reason ! able. 2002.

[43] Timothy van Gelder. The rationale for rationale. *Law, Probability and Risk*, 6:23–42, 2007.

[44] Timothy van Gelder. Argument mapping. In H. Pashler, editor, *Encyclopaedia of the Mind*, volume 1, pages 51–53. Sage, Thousand Oaks, CA, 2013.

[45] Li Wang, Xinya Chen, Chung Wang, Lingna Xu, Rustam Shadiev, and Yan Li. Chatgpt's capabilities in providing feedback on undergraduate students' argumentation: a case study. *Thinking Skills and Creativity*, 2023.

[46] Abdullah Al Zubaer, Michael Granitzer, and Jelena Mitrović. Performance analysis of large language models in the domain of legal argument mining. *Frontiers in Artificial Intelligence*, 6, 2023.

[47] Albert Ĺaszló Barabási. Network science. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 371, 2016.

A An Example of a Processed Argumentative Essay

Original text: *Reducing the usage of cars in today's world could be extremely beneficial. Sure, it is also a hastle having to re-route your commute and also making the time to get there, but the myriad advantages to the reduction of car usage is astonishing. Reducing our usage of cars will reduce the smog in cities, such as Los Angeles, Beijing, and Paris, reduce the stress of many drivers, and also save people money. Now who doesn't love money? Smog/pollution is growing daily in dense, polluted areas. Looking outside of LA, you can see the dirty, polluted air surrounding the city, as well as the toxic blanket the covers Beijing. One of the main sources contributing to this is cars. Greenhouse gases are emitted from tailpipes and go straight into the atmosphere. In Europe, exhaust makes up fifteen percent of greenhouse gas emission, and accounts for fifty percent in the United States, FIFTY PERCENT. That's half of the pollution in our country, and an easy solution is to limit car usage. If your car usage is not limitable, perhaps switching to a hybrid car such as a Toyota Prius will make you feel good about contributing to the cause. Some areas, like Bogota, Colombia, participate in a Car-free*

Day. This day is widely celebrated in the area and is infectiously spreading to nearby areas and potentially the world soon enough. This day allows for smog reduction. Cities such as Paris, however, have to ban car usage sometimes because their smog is so bad. During this ban, hybrid cars and carpooling is allowed. This shows the extreme measures necessary to reduce the smog in populated areas. As most people know, driving is stressful and is perhaps a top contributer of stress in America. In populated areas, rush-hour traffic is annoying and causes many people to change their schedules. When driving during rush-hour, you are in constant fear of potentially being cut-off and your risk of being in an accident heightens dramtically. With that being said, what if I told you there was a way to completely cut out this fear? Communities such as Vauban, Germany are helping alleviate stress by making car-free communities. Within these communities, cars are allowed to be owned, but you must park it in a parking garage at the end of the community and also buy a spot... for $40,000. Cars are used rarely, as restaurants, shops, and others are within walking distance of these communities. Cars are only used for long-distance travel and are permitted on highways and on the outer edges of the area. The stress is alleviated because you can walk outside, grab your mail, and listen to the birds if want, all without the worry of cars. You don't need to constant check your rear-view mirrors if you're walking to your favorite restaurant. With the introduction of smartphones and the constantly growing usage rate of the internet, people don't need cars to communicate anymore. They can simply go on Facebook, Twitter, Snapchat, Instagram, etc. to communicate. Finally, the reduction of car usage allows for people to save money, and a significant amount too. If you live in a neighborhood such as Vauban, there may be no need for a car at all. In today's world, a good car costs somewhere are 30-40 thousand dollars. Imagine what you could do with that much money. You could get a nice house, or travel to your favorite destination, or anything really. You could send your kid to a nice college! And the drawbacks are small, because everything is within a twenty minute walk. That sounds pretty nice to me.

B Computing Environment

All experiments were conducted using Google Colab Pro, leveraging NVIDIA Tesla T4. The specific Python environment used was based on Google Colab's default runtime, with the following specifications (Ubuntu 20.04, Python Version: 3.10.12). On average, processing each essay took approximately 1 minute and 30 seconds.

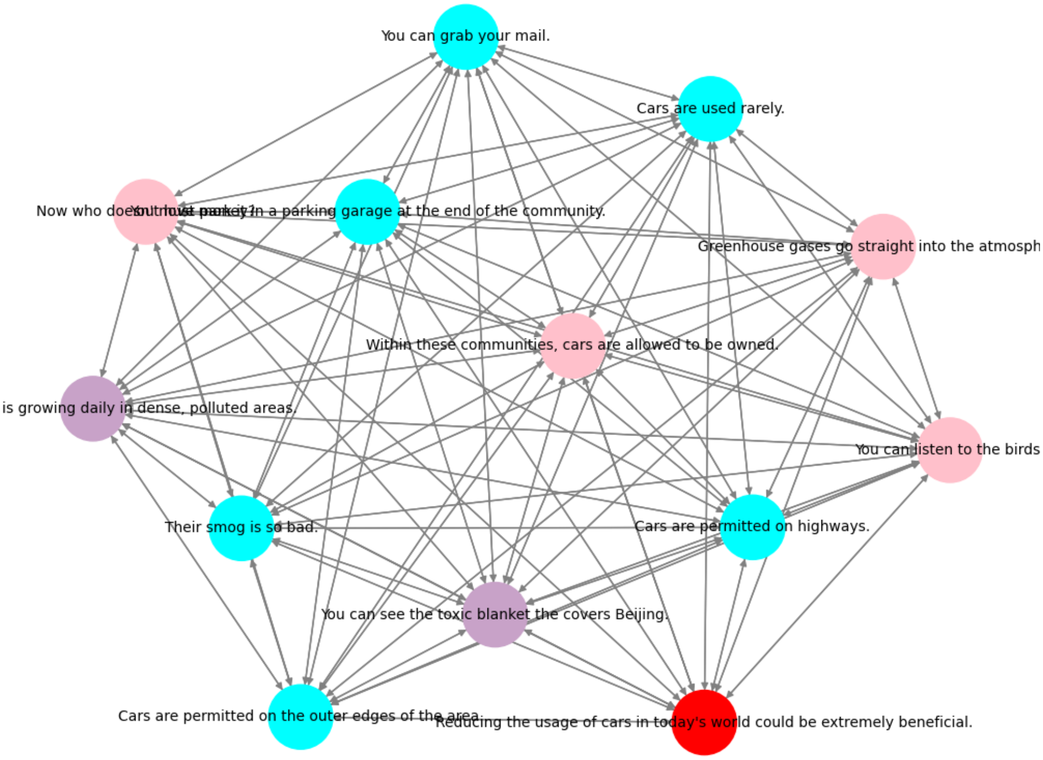
You can grab your mail.
Cars are used rarely.
Now who doesn't love peace?
You must park it in a parking garage at the end of the community.
Greenhouse gases go straight into the atmosph
Within these communities, cars are allowed to be owned.
is growing daily in dense, polluted areas.
You can listen to the birds
Their smog is so bad.
Cars are permitted on highways.
You can see the toxic blanket the covers Beijing.
Cars are permitted on the outer edges of the area.
Reducing the usage of cars in today's world could be extremely beneficial.

Figure 5: Visualization of the graph induced from the essay in the Appendix A. In red the position, in pink the claims, in teal the counterclaims, and in lilac the rebuttals.