Towards a Three-Phase Pipeline for Enthymeme Analysis

Nehal Niloy[1], Federico Castagna[1], Isabel Sassoon[1] and Elfia Bezou-Vrakatseli[2]

[1]Dept of Computer Science, Brunel University of London, UK
[2]Université Côte d'Azur, Inria, CNRS, I3S, France

Abstract

Enthymemes are arguments containing implicit premises or conclusions whose recovery is necessary for complete interpretation. While recent work has explored enthymeme detection and reconstruction, the internal reasoning process remains largely underexplored. In this paper, we present a preliminary three-phase evaluation pipeline for enthymeme modelling consisting of (i) enthymeme detection, (ii) gap localisation, and (iii) enthymeme reconstruction. Unlike previous approaches, the framework explicitly models localisation as a separate stage between detection and generation. Preliminary experiments using transformer-based classifiers and an instruction-tuned language model indicate that the proposed decomposition is feasible and provides a structured basis for analysing reasoning failures in argumentative text.

Keywords

enthymemes, computational argumentation, argument mining, reasoning, large language models

1. Introduction

Human arguments frequently omit information that speakers expect audiences to infer. Such incomplete arguments, known as *enthymemes* [1, 2, 3], play an important role in persuasion and everyday reasoning, and have attracted sustained interest across informal logic, argument mining, and, more recently, large language model (LLM) research [4, 5, 6]. From a computational perspective, however, these omissions create challenges [7] because systems must identify missing information before they can fully analyse argumentative structure, whether that structure is recovered through data-driven generation or through symbolic, logic-based decoding [8, 9, 10].

This paper makes three contributions to computational argumentation and enthymeme modelling. First, we introduce a structured three-phase evaluation framework for enthymeme analysis, decomposing the task into (i) detection of incomplete arguments, (ii) localisation of the missing argumentative component, and (iii) reconstruction of the omitted content. While prior work has primarily treated enthymeme modelling as a combination of detection and reconstruction tasks [11], our framework explicitly formalises gap localisation as an intermediate and independently evaluable stage. This decomposition enables more fine-grained analysis of system behaviour by isolating failure points across classification and generation subtasks. Second, we implement enthymeme modelling as a sequence of task-specific learning problems aligned with distinct data representations and model classes. In particular, we show how binary classification, structured component identification, and conditional text generation can be integrated within a unified pipeline while preserving phase-specific evaluation criteria. This design allows the same underlying argumentative data to be transformed into multiple supervised learning formulations, enabling controlled experimentation across subtasks rather than relying on end-to-end outputs alone. Third, we present an empirical evidence demonstrating the performance of the three stages of this decomposition (detection, localisation and reconstruction) within the proposed framework (Table 2 and Table 3). These results provide initial evidence that supports the feasibility of staged evaluation in enthymeme modelling.

By isolating failure points at each phase, this decomposition is intended to provide a more transparent basis for analysing where computational models succeed or fail when processing incomplete arguments.

CMNA'26: 26th Workshop on Computational Models of Natural Argument, September 2026, Barcelona, Spain
Nehal.Niloy@brunel.ac.uk (N. Niloy)
0009-0001-3495-0405 (N. Niloy); 0000-0002-5142-4386 (F. Castagna); 0000-0002-8685-1054 (I. Sassoon); 0009-0004-8954-246X (E. Bezou-Vrakatseli)

2. Background and Related Work

Within the field of argumentation, enthymeme is understood as an incomplete argument in which a premise or, less often, a conclusion is omitted but intended to be recoverable from context or common knowledge [1, 3, 2]. In Aristotelian terms, enthymemes are compressed syllogistic arguments whose persuasive force depends on the audience supplying an unstated step [1]. Computationally, this means that enthymemes are not instances of missing text, but instances of missing argumentative structure [4]. This matters because explicit representations of premises and conclusions are central to computational argumentation. Whether arguments are represented as labelled spans, structured tuples, or graph-based frameworks, missing components reduce the ability of a system to evaluate support relations, detect inconsistencies, or explain why a conclusion follows [5, 8]. In practical natural language processing (NLP) settings, a system may process the surface form of argument fluently while still failing to account for the inferential step that makes the argument intelligible. Accordingly, others have treated enthymeme modelling as an important bridge between natural language argument mining and more formal models of reasoning [12]. Other NLP work on enthymemes has instead focused on two tasks: *detection*, i.e. identifying whether an argument contains a missing argumentative discourse unit, and *reconstruction*, i.e. generating the omitted content [6]. Stahl et al. introduced a large-scale synthetic corpus based on learner essays by deleting central argumentative units, thereby enabling supervised training for both tasks [11]. Their work established a benchmark on enthymeme modelling and showed that these problems can be cast in terms of modern transformer classification and generation models.

A complementary line of work approaches enthymeme decoding from a symbolic, logic-based perspective rather than a data-driven generation perspective. Ben-Naim et al. [9] propose an axiomatic framework for evaluating candidate decodings of enthymemes in weighted structured argumentation, introducing criteria for judging decoding quality in cases where several logically valid completions exist. David and Hunter [10] similarly formalise enthymeme decoding using default logic over argument maps, casting the choice among candidate decodings as a MaxSAT optimisation problem that must respect the support and attack relations of the map. These approaches are naturally positioned alongside our Phase 3 (Reconstruction): whereas our pipeline learns to generate the missing component directly from data using an instruction-tuned language model, the axiomatic and logic-based criteria proposed in [9, 10] offer complementary, verifiable notions of what constitutes a *good* decoding. Incorporating such criteria to filter, re-rank, or evaluate the candidates generated by our Phase 3 model is a promising direction that we discuss in Section 6.

3. Three-Phase Pipeline

The pipeline analyses incomplete arguments in three successive phases: detection, gap localisation, and reconstruction. Before a model can reconstruct a missing argumentative step, it must first identify that a gap is present and determine what kind of component is absent. Each phase therefore addresses a distinct question and is trained separately, as shown in Figure 1.

- **Phase 1: Enthymeme Detection**. This stage is formulated as a *binary classification task* with labels consisting of 0 (argument) and 1 (enthymeme). Given a (supposed) argument, the model predicts whether it is an enthymeme. For example, the argument "There is smoke coming from that building, so there must be a fire." would be labelled as 1 as it is an enthymeme missing the premise "Smoke is caused by fire".

- **Phase 2: Gap Localisation**. Once incompleteness has been detected, the second phase identifies which argumentative component is absent. We employ three labels:

  Missing Major Premise, Missing Minor Premise, Missing Conclusion. This choice is motivated by the classical syllogistic view of argument associated with Aristotle, where a conclusion follows from the interaction of a general rule-like premise and a case-specific premise [1]. Although

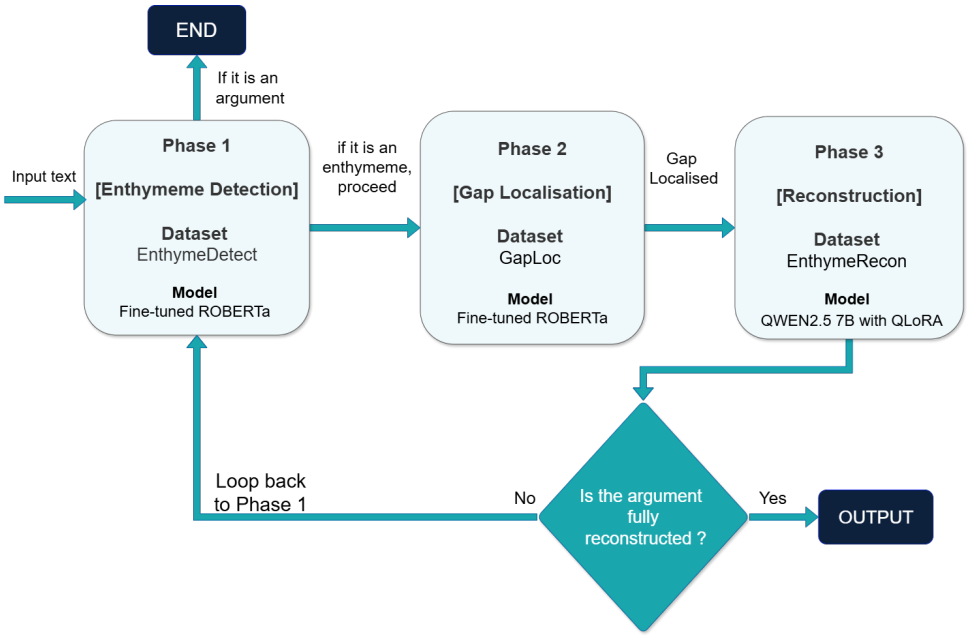

Figure 1: Three-phase enthymeme analysis pipeline. Gap localisation (Phase 2) is an explicit, independently evaluable stage between detection and reconstruction. The completeness-check loop back to Phase 1 and goes through the pipeline until its detected as an argument.

natural arguments are often more flexible than canonical syllogisms, this structure provides a clear and operational representation for identifying where an enthymematic gap occurs.

Introducing localisation as an explicit stage enables finer-grained analysis of reasoning behaviour than detection alone. To illustrate why this stage matters, consider again the enthymeme "There is smoke coming from that building, so there must be a fire". Without an explicit localisation step, a reconstruction model has no signal indicating *which* structural component is missing, and may generate a plausible-sounding but structurally inappropriate completion, for instance, a redundant restatement of the conclusion ("The building is on fire") or an unrelated case-specific fact, rather than the missing general rule ("Smoke is caused by fire"). By first predicting the label *Missing Major Premise*, Phase 2 directly conditions Phase 3 on the correct component type before generation begins, removing this ambiguity. The reconstruction results in Section 5 (Table 3) provide empirical support for this: reconstruction quality drops substantially when this conditioning signal is withheld.

- **Phase 3: Reconstruction**. The final phase reconstructs the missing argumentative component. Given the remaining parts of the argument, an instruction-tuned language model generates the omitted premise or conclusion. The reconstruction should not be declared complete merely because the generator has produced fluent text or obtained a high n-gram-overlap score. For the present tripartite representation (more in section 4.1), a candidate is structurally complete when it contains a non-empty generated component of the role predicted by Phase 2, together with the two observed components, so that all three slots are filled. A stronger claim of adequate reconstruction requires at least three further checks: the generated component must match the requested role, be compatible with the other two components, and provide an inferential link that is acceptable in context.

  In our running example, the model receives the enthymeme consisting of the minor premise "Socrates is a man" and the conclusion "Socrates is a mortal", together with an instruction to reconstruct the *major premise*, and should generate "All men are mortal".

4. Experimental Setup

For training and evaluation, the pipeline reuses the same underlying source data across all three phases, reformatted into a different configuration for each phase, because every phase requires a different representation of argumentative structure. The central design principle is that the data format should reflect the learning objective of the phase. To this end, the pipeline uses the datasets *EnthymeDetect*,[1] for Phase 1, *GapLoc*,[2] for Phase 2 and *EnthymeRecon*,[3] for Phase 3, all of which are curated versions of the $EthiX_e$ dataset [13]. $EthiX_e$ is a subset of the *EthiX* dataset [14], which contains manually annotated arguments extracted from Kialo,[4] a user-generated platform with structured and moderated debates. $EthiX_e$ contains only those arguments from *EthiX* that are enthymematic. Each instance (i.e., each entry in the dataset) consists of a short natural language argument, the central question of the debate from which the argument was extracted, and an argument-scheme label. The dataset was also annotated using Argument Scheme Key (ASK), an approach proposed by Visser et al. [15] as an effective methodology for choosing the appropriate scheme from Walton's taxonomy [16]. The $EthiX_e$ material spans multiple argumentative domains, including environment, ethics, health, policy, economics, education, social media, cloning, and artificial general intelligence, thus covering a broad range of topics.

4.1. Dataset Validation and Reliability Metrics

In its original form, $EthiX_e$ is not directly suitable for localisation or reconstruction, because it does not explicitly separate arguments into distinct structural components such as premises and conclusion. For this reason, the Phase 2 and 3 datasets were crafted following the set rules: the general rule (major premise), the case-specific statement (minor premise), and the conclusion (as a bridge connecting major and minor) while preserving the meaning of the original argument.

Ambiguous cases were resolved through discussion between authors. Applying this procedure across the dataset converted the enthymematic instances into a common tripartite representation consisting of *major premise*, *minor premise*, and *conclusion*. This canonicalisation was introduced so that the later phases could be formulated as supervised learning tasks with a shared structural label space.

LLM-as-a-Judge	Syllogistic Soundness	Structural Alignment	Linguistic Naturalness
GapLoc + EnthymeRecon samples	0.78	0.88	4.77

Table 1
LLM-as-a-Judge Validation Averages

To explicitly validate the logical boundaries of the manually annotated tripartite dataset, an automated LLM-as-a-Judge assessment [17] was implemented using Gemini (gemini-2.5-flash)[5]. The judge evaluated a representative sample of the human-constructed triples across three core dimensions: Syllogistic Soundness (binary deduction verification), Structural Alignment (canonical taxonomic compliance), and Linguistic Naturalness (1–5 Likert scale). Table 1 reports that the evaluation yielded a high Linguistic Naturalness score of 4.77 / 5, confirming that the manual reconstruction preserved high fluid phrasing suitable for downstream natural language understanding. The model registered a Structural Alignment score of 0.88 out of 1, demonstrating that the categorical taxonomy of Major Premise (generalization) and Minor Premise (instantiation) was cleanly preserved during manual separation. Crucially, the Syllogistic Soundness metric was recorded at 78%. Rather than indicating annotation error, this baseline score empirically confirms a well-documented challenge in computational argumentation: naturally occurring human enthymemes extracted from real-world debates (such as Kialo) frequently rely on

[1] https://huggingface.co/datasets/Nehal02/Pipeline/blob/main/EnthymeDetect.csv
[2] https://huggingface.co/datasets/Nehal02/Pipeline/blob/main/GapLoc.csv
[3] https://huggingface.co/datasets/Nehal02/Pipeline/blob/main/EnthymeRecon.csv
[4] https://www.kialo.com/
[5] https://ai.google.dev/gemini-api/docs/models/gemini-2.5-flash

presumptive or inductive reasoning schemes rather than strict formal-logical deduction. This metric provides a transparent, quantifiable boundary of the dataset's logical structure [11].

4.2. Models and Implementation

The implementation choices in this work are guided by the functional demands of each phase rather than by a single uniform architecture. Since Phases 1 and 2 are both classification problems over relatively short argumentative inputs, they require models that produce strong contextual sentence representations and stable decision boundaries. For this reason, RoBERTa-based[6] sequence classifiers were used for detection and localisation. It is a robust encoder architecture designed for sentence and sequence-level classification tasks, making it appropriate for deciding whether an argument is incomplete and, if so, which component is missing [18]. It is also an economical design choice for the present study as RoBERTa is open-source, comparatively lightweight to fine-tune, cheap and straightforward to run in standard notebook environments such as Google Colab. Phase 3 differs substantially from the first two phases because it is a conditional text generation task rather than a classification task. Here the model must produce a linguistically plausible and structurally appropriate missing premise or conclusion from the remaining argument context. Therefore Qwen2.5-7B-Instruct[7], an instruction-tuned language model, was used. Qwen2.5-7B is an open-source model that demonstrates strong benchmark performance for its size, offering a favourable balance between generation quality and computational cost [19]. It is comparatively cost-effective to fine-tune, and is highly optimised for parameter-efficient adaptation. This model features architectural optimisations suited for structured text generation and logical parsing. To make fine-tuning computationally efficient given the model's scale, we employed Quantized Low-Rank Adaptation (QLoRA) [20]. More broadly, this division of models reflects a methodological commitment of the pipeline: each phase is matched with an architecture suited to its task formulation. The pipeline is also not tied to these two specific models (RoBERTa, Qwen2.5-7B) rather, it is intended to be model-agnostic in principle: alternative encoder models could be substituted in the classification phases, and alternative pre-trained or instruction-tuned models could be used in the reconstruction phase, provided that they suit the task formulation and computational constraints.

4.3. Evaluation metrics

Detection and gap localisation are evaluated using Accuracy, Precision, Recall, and F1-score. While Accuracy measures overall correctness, Precision and Recall capture false-positive and false-negative behaviour respectively, and F1-score provides their harmonic mean. Reconstruction is evaluated using BLEU and ROUGE metrics. BLEU measures n-gram overlap between generated and reference reconstructions, while ROUGE-1, ROUGE-2, and ROUGE-L evaluate unigram, bigram, and sequence-level similarity [21, 22]. These metrics provide an initial assessment of generation quality.

5. Preliminary Results

Table 2 reports Phase 1 and Phase 2 results, and Table 3 reports Phase 3 results; all figures are computed on the held-out test set of the respective dataset (*EnthymeDetect*, *GapLoc*, and *EnthymeRecon*).

Phase	Accuracy	Precision	Recall	F1
Detection	0.91	0.93	0.92	0.92
Gap Localisation	0.94	0.95	0.96	0.94

Table 2
Preliminary Phase 1 and 2 results.

Configuration	BLEU	R1	R2	RL
With Localisation	0.88	0.89	0.92	0.92
Without Localisation	0.48	0.65	0.56	0.77

Table 3
Preliminary Phase 3 results.

[6] https://huggingface.co/FacebookAI/xlm-roberta-base/
[7] https://huggingface.co/Qwen/Qwen2.5-7B-Instruct/

The results suggest that enthymeme detection and localisation can be learned reliably using transformer-based classifiers. In particular, the strong performance of Phase 2 indicates that the structural role of a missing component can often be inferred directly from the remaining argument context.

The comparison suggests that explicit gap localisation substantially benefits downstream reconstruction quality. This supports the inclusion of Phase 2 as an independent stage rather than treating reconstruction as a direct generation task [11].

We note that the absolute BLEU and ROUGE values obtained here, particularly in the *With Localisation* configuration, are high relative to what is typically reported for open-ended argument generation. We attribute this partly to the relatively small size and syllogistic structure of *EnthymeRecon*, where major premises in particular are often expressed using a limited set of recurring general-rule constructions, and partly to the modest scale of the current test set. These preliminary results should therefore be read as an initial feasibility signal for the value of explicit localisation, rather than as a mature estimate of reconstruction quality on more diverse, naturally occurring enthymemes.

6. Discussion and Conclusion

The principal contribution of this work is methodological. Rather than framing enthymeme analysis as a single end-to-end task, the proposed framework decomposes it into three interpretable phases, enabling analysis of failure points and providing a basis for evaluating reasoning capabilities in both traditional NLP models and large language models. Preliminary experiments demonstrate that each stage: detection, gap localisation, and reconstruction can be implemented successfully using modern transformer architectures and instruction-tuned language models, and that explicit gap localisation improves reconstruction quality relative to a direct generation baseline. This further corroborates studies showing that argumentative methods can support AI natural language understanding, thereby enhancing their reasoning capabilities [23, 24].

Several aspects of the pipeline remain preliminary and motivate concrete next steps. Firstly, further evaluation of the reconstruction phase, as generation is arguably the most difficult and most consequential part of enthymeme modelling, future work should prioritise completing this evaluation using both automatic and human centred criteria [11], for instance via a further classifier, an entailment check, or the axiomatic and logic-based decoding criteria of enthymeme [9, 10]. Second, as noted in Section 5, the current reconstruction scores were obtained on a small, manually curated tripartite dataset, and further work is needed to assess how well they generalise to more diverse, naturally occurring enthymemes; this includes human evaluation of reconstruction quality and extension to richer argumentative structures beyond the tripartite representation. Third, the three phases were evaluated independently in this paper; future work will report end-to-end pipeline performance, chaining Phases 1-3 on unseen enthymemes, together with an ablation study that isolates the specific contribution of Phase 2 to overall pipeline accuracy, complementing the phase-wise results reported here.

In summary, we presented a preliminary three-phase evaluation pipeline for enthymeme analysis consisting of detection, gap localisation, and reconstruction. The framework offers a transparent approach for studying implicit reasoning in incomplete arguments and provides a foundation for future research on argument-aware NLP systems that combines data-driven generation with the symbolic evaluation criteria emerging in the computational argumentation literature.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] Aristotle, Rhetoric, volume 11, Modern Library, New York, 1954.

[2] D. Walton, Enthymemes, common knowledge, and plausible inference, Philosophy & rhetoric 34 (2001) 93–112.
[3] D. Hitchcock, Enthymematic arguments, in: On Reasoning and Argument: Essays in Informal Logic and on Critical Thinking, Springer, 2017, pp. 39–56.
[4] N. Niloy, F. Castagna, I. Sassoon, Enthymemes in Large Language Models: A Survey, ResearchGate preprint 10.13140/RG.2.2.32429.35044 (2025).
[5] J. Lawrence, C. Reed, Argument Mining: A Survey, Computational Linguistics 45 (2019) 765–818.
[6] E. Sviridova, E. Cabrio, S. Villata, Mining implicit arguments for reasoning: A survey, Argument & Computation (2025) 19462174251344764.
[7] J. Aquino, F. Castagna, I. Sassoon, A survey on reasoning in large language models: Issues, impacts, and improvements, ResearchGate preprint 10 (2025).
[8] J. Ben-Naim, V. David, A. Hunter, Understanding Enthymemes in Argument Maps: Bridging Argument Mining and Logic-based Argumentation, ArXiv abs/2408.08648 (2024). URL: https://doi.org/10.48550/arXiv.2408.08648.
[9] J. Ben-Naim, V. David, A. Hunter, An Axiomatic Study of a Modular Evaluation of Enthymeme Decoding in Weighted Structured Argumentation, in: Proceedings of the 22nd International Conference on Principles of Knowledge Representation and Reasoning (KR'25), 2025, pp. 110–120. doi:10.24963/kr.2025/11.
[10] V. David, A. Hunter, A Logic-based Framework for Decoding Enthymemes in Argument Maps Involving Implicitness in Premises and Claims, in: Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI'25), 2025, pp. 4445–4453. doi:10.24963/ijcai.2025/495.
[11] M. Stahl, N. Düsterhus, M.-H. Chen, H. Wachsmuth, "Mind the Gap: Automated Corpus Creation for Enthymeme Detection and Reconstruction in Learner Arguments", in: H. Bouamor, J. Pino, K. Bali (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2023, Association for Computational Linguistics, Singapore, 2023, pp. 4703–4717. URL: https://aclanthology.org/2023.findings-emnlp.312/. doi:10.18653/v1/2023.findings-emnlp.312.
[12] A. Xydis, F. Castagna, E. Sklar, S. Parsons, Applications of argumentation-based dialogues, Journal of Applied Logics 3 (2025) 427– 487.
[13] E. Bezou-Vrakatseli, O. Cocarascu, S. Modgil, Can Large Language Models Understand Argument Schemes?, in: Findings of the Association for Computational Linguistics: ACL 2025, 2025, pp. 13666–13681.
[14] E. Bezou-Vrakatseli, O. Cocarascu, S. Modgil, EthiX: A Dataset for Argument Scheme Classification in Ethical Debates, in: U. Endriss, et al. (Eds.), Proceedings of the 27th European Conference on Artificial Intelligence (ECAI), volume 392, 2024, pp. 3628–3635.
[15] J. Visser, J. Lawrence, C. Reed, J. Wagemans, D. Walton, Annotating Argument Schemes: J. Visser et al., Argumentation 35 (2021) 101–139.
[16] F. Macagno, Argumentation schemes, The Routledge Handbook of Argumentation Theory (2025) 169–182.
[17] X. Li, R. Zhang, I. Gurevych, LLM-as-Judge: A Survey of Methods, Applications, and Limitations, Journal of Artificial Intelligence Research 74 (2024) 321–348.
[18] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, RoBERTa: A Robustly Optimized BERT Pretraining Approach, ArXiv abs/1907.11692 (2019). URL: https://doi.org/10.48550/arXiv.1907.11692.
[19] Q. A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, G. Dong, H. Wei, H. Lin, J. Yang, J. Tu, J. Zhang, J. Yang, J. Yang, J. Zhou, J. Lin, K. Dang, K. Lu, K. Bao, K. Yang, L. Yu, M. Li, M. Xue, P. Zhang, Q. Zhu, R. Men, R. Lin, T. Li, T. Xia, X. Ren, X. Ren, Y. Fan, Y. Su, Y.-C. Zhang, Y. Wan, Y. Liu, Z. Cui, Z. Zhang, Z. Qiu, S. Quan, Z. Wang, Qwen2.5 Technical Report, ArXiv abs/2412.15115 (2024). URL: https://doi.org/10.48550/arXiv.2412.15115.
[20] T. Dettmers, A. Pagnoni, A. Holtzman, L. Zettlemoyer, QLoRA: Efficient Finetuning of Quantized LLMs, https://arxiv.org/abs/2305.14314 (2023).
[21] C.-Y. Lin, ROUGE: A Package for Automatic Evaluation of Summaries, Text Summarization

Branches Out: Proceedings of the ACL-04 Workshop (2004) 74–81. URL: https://aclanthology.org/W04-1013/.

[22] K. Papineni, S. Roukos, T. Ward, W.-J. Zhu, Bleu: A Method for Automatic Evaluation of Machine Translation, in: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, 2002, pp. 311–318.

[23] F. Castagna, I. Sassoon, S. Parsons, Can formal argumentative reasoning enhance LLMs performances?, arXiv preprint arXiv:2405.13036 (2024).

[24] F. Castagna, N. Kökciyan, I. Sassoon, S. Parsons, E. Sklar, Computational argumentation-based chatbots: a survey, Journal of Artificial Intelligence Research 80 (2024) 1271–1310.