Neurosymbolic Learning for Inference-Time Argumentation

Gabriel Freedman* **Adam Dejl*** **Adam Gould*** **Mansi***
Lihu Chen **Junqi Jiang** **Francesca Toni**
Department of Computing, Imperial College London
{g.freedman22, ad5518, amg2118, m24}@imperial.ac.uk
{junqi.jiang, lihu.chen, ft}@imperial.ac.uk

Abstract

Claim verification is an important problem in high-stakes settings, including health and finance. When information underpinning claims is incomplete or conflicting, uncertain answers may be more appropriate than binary true or false classifications. In all cases, faithful explanations of the considerations determining the final verdict are crucial. We introduce *inference-time argumentation (ITA)*, a trainable neurosymbolic framework for ternary claim verification in which a formal argumentation semantics giving the strength of claims and arguments is used both (i) to guide LLM training as models learn to *generate arguments* for and against claims and *assign them base scores* (representing intrinsic strengths) and (ii) to compute ternary (true/false/uncertain) predictions from generated, scored arguments. As a result, at training time, argument generation and scoring can be optimised according to the quality of the induced argumentative predictions. Moreover, at inference time, the final prediction is faithful, by construction, to the arguments and scores determining the verdict, rather than being justified by a potentially unfaithful post-hoc reasoning trace as in conventional reasoning models. We finally show that, on two datasets for ternary claim verification, ITA improves upon argumentative baselines and can perform competitively against non-argumentative direct-prediction baselines, while providing verdicts that are computed deterministically from explicit, inspectable argumentative structures. Our code, models and data are available at `https://github.com/CLArg-group/inference-time-argumentation`.

1 Introduction

Claim verification amounts to ascertaining the veracity of textual claims. It is a widely studied problem, recently as a potential application for LLMs [Dmonte et al., 2024], and is used in many high-stakes settings, including healthcare [Vladika et al., 2025], finance [Zhao et al., 2024b] and scientific discovery [Alvarez et al., 2024]. In all settings, information underpinning the claims is crucial in ascertaining their veracity, but when this information is incomplete or conflicting, *uncertain answers* may be more appropriate than true or false classifications. For example FEVER [Thorne et al., 2018] uses Supported, Refuted, and NotEnoughInfo labels, while AVeriTeC [Schlichtkrull et al., 2023] distinguishes between Supported, Refuted, Not Enough Evidence, and Conflicting Evidence/Cherry-picking. No matter the veracity labels, *faithful explanations* of the considerations determining the labels are crucial in claim verification [Freedman et al., 2025].

In this paper we consider claim verification as a three-valued prediction problem. Given a natural language claim, we aim to automatically predict a verdict $\hat{y} \in \{\textit{True}, \textit{False}, \textit{Uncertain}\}$. The *Uncertain* label is not treated as failure to classify, but as a substantive verdict for cases in which the

*Equal contribution.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

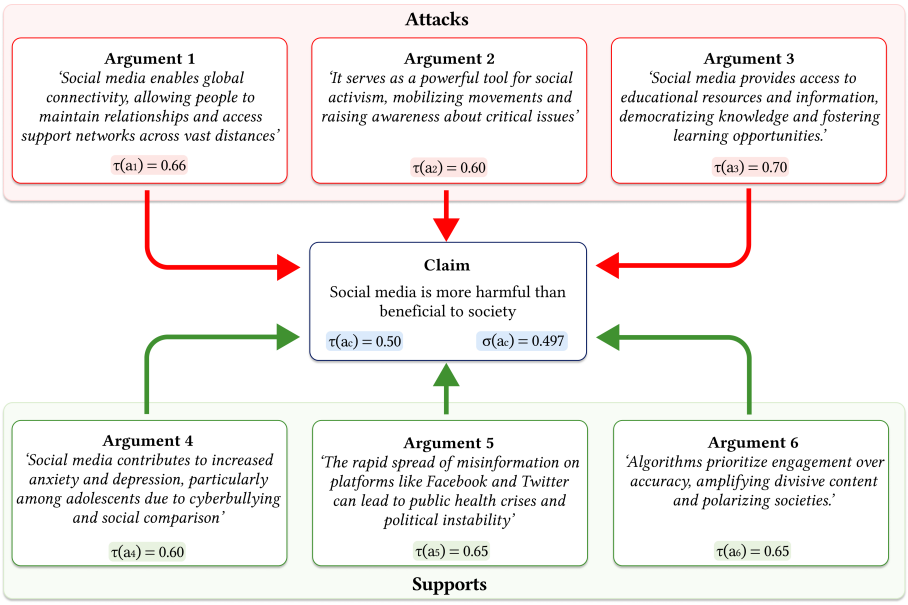

Figure 1: An example argumentative structure for a claim adapted from DEBATunE [Li et al., 2024a]. Arguments 1-3 attack the claim and 4-6 support it. For claim/argument x, $\tau(x)$ is the base score and $\sigma(x)$ the argumentative strength computed by the DF-QuAD semantics [Rago et al., 2016]. ITA classifies this claim as *Uncertain* as its strength is close to the midpoint of [0,1].

available information does not warrant accepting or rejecting the claim, as in Figure 1. To support this form of ternary claim verification while also guaranteeing faithful explanations for veracity verdicts, we introduce *inference-time argumentation (ITA)*, a novel neurosymbolic framework for training LLMs to argue about the veracity of claims. Rather than treating veracity outcomes as labels to be predicted directly, ITA derives them from the formal evaluation of generated supporting and attacking arguments, making prediction an argumentative verdict rather than a bare classification. In ITA, similarly to ArgLLMs [Freedman et al., 2025], verdicts are obtained by applying thresholds to a continuous *argumentative strength* computed with a (deterministic) argumentation semantics [Rago et al., 2016]. The final verdicts are thus derived based on the argumentative structure of supporting and attacking arguments and the associated *base scores* for all arguments and the claim, representing their intrinsic strength. As in ArgLLMs, the argumentative structure serves as a faithful explanation for the verdict. However, rather than using off-the-shelf LLMs for generating and scoring arguments as ArgLLMs do [Freedman et al., 2025, Zhou et al., 2025], ITA is based on training them so that LLMs learn how to generate and/or score arguments optimally for the downstream prediction. Thus, in ITA, the argumentation semantics also informs training. Specifically, it supervises the arguments' base score generation and provides feedback for argument generation with reinforcement learning (RL). In other words, ITA is neurosymbolic in combining neural argument generation and scoring with symbolic reasoning over QBAFs. The formal argumentation semantics determines the verdict and provides the learning signals for the neural components.

ITA differs from (non-argumentative) inference-time reasoning approaches based on reinforcement learning from verifiable rewards [Guo et al., 2025] in its focus on a task where outcomes may be uncertain, rather than exact and automatically checkable with certainty (e.g. as in mathematics or programming), and in its guarantees of faithful explainability, rather than with reasoning traces understood post-hoc as explanations, with several documented limitations [Chen et al., 2025, Cornish and Rogers, 2025, Yu et al., 2026].

In summary, our contributions are as follows. First, we introduce ITA, a trainable neurosymbolic framework for three-valued claim verification in which verdicts are computed from argumentative structures (see Figure 2). Second, we introduce two training strategies for ITA, respectively based on (i) learning base scores for generated arguments through objectives defined over the downstream argumentative verdict, rather than relying only on base scores provided by base LLMs as in Freedman et al. [2025], and (ii) optimising the argument generator using feedback induced by formal

argumentation semantics, turning argumentation from an inference-time decision layer into a training signal. Finally, we evaluate ITA on two claim-verification datasets against non-argumentative direct-prediction baselines and argumentative baselines, showing that we outperform all faithfully explainable baselines while performing competitively against others. We obtain the datasets by adapting three human-curated datasets: AVeriTeC [Schlichtkrull et al., 2023], StrategyClaim [Freedman et al., 2025] and DEBATunE [Li et al., 2024a].

2 Related Work

Abstention. Several LLM-based claim verification methods exist [Dmonte et al., 2024]. Here we focus on methods aware of their knowledge boundaries, that honestly say "I do not know" for non-answerable questions, known as abstention [Li et al., 2024b, Wen et al., 2025]. Existing approaches teach LLMs to perform abstention through fine-tuning strategies [Amayuelas et al., 2024, Kapoor et al., 2024]. Cohen et al. [2024] further improve this line of research by introducing a special `[IDK]` ("*I don't know*") token into the model's vocabulary. While these approaches all focus on training models to refuse to answer certain queries, they do not provide faithful explanations for why these refusals are the appropriate response. Nor do they seek to explicitly surface their reasoning when conflicting evidence, rather than just lack of knowledge, is the cause for their uncertainty.

Learning and Argumentation. Argumentation has been advocated in combination with learning in several settings (as recently surveyed by Proietti and Toni [2023], Rago et al. [2024]). In the context of RL, Gao and Toni [2014] use reasoning with argumentation frameworks to shape rewards, in a multi-agent simulated game setting, and Ward et al. [2022] replace human feedback in RL with reasoning in preference-based argumentation, in a maze environment. Neither of these approaches generates arguments or uses gradual argumentation semantics as we do. In the context of neurosymbolic, end-to-end learning, Gould and Toni [2025] adopt gradual argumentation semantics as well as, like in our approach, base score model learning. However, in their approach arguments are given rather than generated by LLMs, and no RL takes place.

LLMs and Argumentation. Closest to our work are the proposals by Freedman et al. [2025] and Zhu et al. [2025], both using argumentation in combination with LLMs, and highlighting its benefits in terms of faithfulness of explanations. However, both approaches focus on generation of arguments and base scores from off-the-shelf LLMs rather than training them. Also, both focus on binary rather than ternary claim verification. Further, Vasileiadis et al. [2025] use a transformer-based encoder for feature extraction as an input to a symbolic learner for argumentation frameworks. However, the resulting neurosymbolic approach is not end-to-end. Moreover, the learning does not make use of argumentation semantics, focusing on a different form of (structured) argumentation.

Explainable Claim Verification. PACAR [Zhao et al., 2024a] combines claim decomposition, dynamic planning and specialised reasoning actions, while LoCal [Ma et al., 2025] uses agents for decomposition, reasoning and logical and causal evaluation. Both provide interpretable claim verification grounded in external evidence. However, their decision procedures still rely on LLM judgments over the generated reasoning. In ITA, the final verdict follows deterministically from the argumentative structure and base scores under the chosen formal semantics and thresholds. This guarantees procedural faithfulness by construction. We leave the integration of external evidence as future work.

3 Background

Quantitative Bipolar Argumentation Frameworks (QBAFs) and Argumentation Semantics. QBAFs [Baroni et al., 2019] are graphical argumentative structures from the field of Argumentation [Atkinson et al., 2017], in which arguments are nodes and relations are edges representing support or attack relations, and arguments are assigned intrinsic strengths (referred to as *base scores*). Formally, a QBAF is $Q = \langle A, R^-, R^+, \tau \rangle$, where A is a finite set of arguments, $R^- \subseteq A \times A$ is an attack relation, $R^+ \subseteq A \times A$ is a support relation, and $\tau : A \to [0, 1]$ assigns each argument a base score ($\langle A, R^-, R^+ \rangle$ alone is called a BAF). Intuitively, $(a, b) \in R^+$ means that argument a supports argument b, and $(a, b) \in R^-$ means that argument a attacks argument b. Argumentation semantics assign *(argumentative) strengths* to the arguments in a QBAF by aggregating their base scores with the effects of their supporters and attackers. Formally, for a QBAF Q, an argumentation semantics is

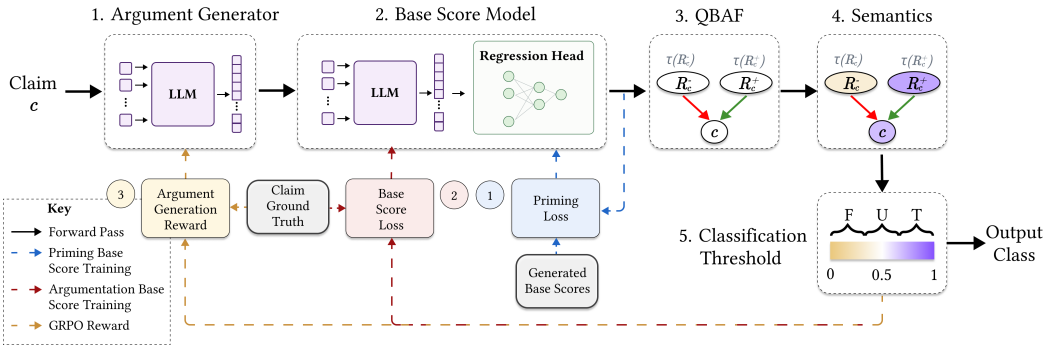

Figure 2: Overview of inference-time argumentation. Given a claim, the argument generator produces supporting and attacking arguments about the claim. A base score module assigns intrinsic strengths to the generated arguments. These steps induce a QBAF centred on the claim, which receives a fixed neutral base score of 0.5. The regression head of the base score model can be primed using generated base scores during training (training phase ①). An argumentation semantics produces a final claim strength, which is thresholded into *True*, *False*, or *Uncertain*. During training, the same semantic evaluation informs a weakly-supervised loss for training the base score model (training phase ②) and argumentation-based reward for the argument generator (training phase ③).

a function $\sigma_Q : A \to [0, 1]$. Different semantics encode different assumptions about how support and attack should interact. In the main experiments, we use DF-QuAD [Rago et al., 2016] for direct comparison with ArgLLMs which uses it already [Freedman et al., 2025] (see Appendix D for the formal definition).

Argumentative LLMs (ArgLLMs). ArgLLMs [Freedman et al., 2025] instantiate QBAFs and argumentation semantics for binary (*True/False*) claim verification. They use an LLM to construct a QBAF $Q_c = \langle A_c, R_c^-, R_c^+, \tau_c \rangle$ for each given claim c. As arguments are represented in free text, we use the set $\mathcal{T}$ to refer to all possible sequences of input tokens and the set $\mathcal{A} \subseteq \mathcal{T}$ as the set of all possible arguments. Thus, we have that $A_c \subseteq \mathcal{A}$. In ArgLLMs, like in our approach, the claim is represented as an argument $a_c \in A_c$, and generated arguments in $A_c \setminus \{a_c\}$ either support or attack a_c, as illustrated in Figure 1. Then, an argumentation semantics σ is applied to obtain the claim strength $\sigma_Q(a_c)$[2]. A verdict is finally derived from this strength. While ITA keeps the same argumentative structure (QBAFs) and semantics (DF-QuAD) as ArgLLMs, it differs by focusing on QBAFs of depth 1 only (ignoring QBAFs of depth 2) but with any number of attackers and supporters (rather than one attacker and one supporter). Also, while ArgLLMs are training-free, ITA learns how to generate arguments and how to assign base scores to them.

Group Relative Policy Optimisation (GRPO). GRPO [Shao et al., 2024] is a critic-free method for RL fine-tuning of LLMs, introduced as a resource efficient variant of Proximal Policy Optimisation. Given a prompt, the policy samples a group of completions, each of which receives a reward. Rather than learning a separate value model, GRPO estimates relative advantages by comparing rewards within the sampled group. We adapt GRPO to argument generation, using rewards based on semantics.

4 Framework Overview

The ITA framework consists of two trainable components (see Figure 2). The first is an *argument generator*, parameterised by θ, producing arguments supporting and attacking c:

$$D_\theta(c) = (S_c, T_c),$$

where S_c and T_c are the set of generated supporting and attacking arguments, respectively. These arguments induce a BAF $\langle A_c, R_c^-, R_c^+ \rangle$ centred on the claim, c, represented as an argument a_c, with:

$$A_c = \{a_c\} \uplus S_c \uplus T_c, \qquad R_c^+ = \{(a, a_c) : a \in S_c\}, \qquad R_c^- = \{(a, a_c) : a \in T_c\}.$$

The resulting graph represents the reasons offered for and against the veracity of the claim.

[2]In the remainder, for readability, we will often omit the QBAF Q and simply write $\sigma(a_c)$, as in Figure 1.

The second component is a *base score model (BSM)*, parameterised by ϕ, assigning base scores to the generated arguments. This BSM is a learnable function $f_\phi : \mathcal{A} \times \mathcal{A} \times \{0,1\} \to [0,1]$ that takes as input a_c, an argument a and their relationship as given by the indicator function $\mathbf{1}_{a \in S_c}$ (returning 0 for attacks and 1 for supports), and outputs a value in $[0,1]$. The claim argument itself is not scored but instead its base score is fixed to a neutral prior of 0.5. This gives the QBAF

$$Q_c^{\theta,\phi} = \langle A_c, R_c^-, R_c^+, \tau_{\phi,c} \rangle, \qquad \text{with } \tau_{\phi,c}(a) = \begin{cases} 0.5 & a = a_c, \\ f_\phi(a_c, a, \mathbf{1}_{a \in S_c}) & \text{otherwise.} \end{cases}$$

When the parameters are clear from context, we will write Q_c for $Q_c^{\theta,\phi}$ and similarly suppress for readability the dependence on the parameters for both trainable components. An argumentation semantics σ then aggregates the effects of supporting and attacking arguments, given their base scores, to compute the final strength of the claim argument: $\hat{v}_c = \sigma(a_c) \in [0,1]$. This value represents the argumentative strength of the claim after evaluating the generated debate. It is mapped to a three-valued verdict $\hat{y}_c$ using thresholds α and β by means of the following *thresholding* function h:

$$\hat{y}_c = h(\hat{v}_c) = \begin{cases} \textit{False} & \text{if } \hat{v}_c \leq \alpha, \\ \textit{True} & \text{if } \hat{v}_c \geq \beta, \\ \textit{Uncertain} & \text{otherwise.} \end{cases}$$

Thus, *Uncertain* corresponds to an intermediate argumentative strength, arising when the debate represented by the QBAF does not make the claim sufficiently strong to accept or weak to reject.

5 Weakly Supervised Learning for Base Score Attribution

For our trained BSMs, we instantiate f_ϕ as follows:

$$f_\phi(a_c, a, \mathbf{1}_{a \in S_c}) = f_{\phi_{reg}}(f_{\phi_{llm}}(g(a_c, a, \mathbf{1}_{a \in S_c})))$$

We use three components. First, $g : \mathcal{A} \times \mathcal{A} \times \{0,1\} \to \mathcal{T}$ constructs a task-specific prompt[3] from the claim, the argument and their relationship. Second, $f_{\phi_{llm}} : \mathcal{T} \to \mathbb{R}^d$ maps the prompt to a d-dimensional latent space embedding using an LLM. Finally, $f_{\phi_{reg}} : \mathbb{R}^d \to [0,1]$ is a feed-forward regression head that converts the embedding to a scalar score. Intuitively, the prompt generator encodes the scoring task, the LLM produces a contextual embedding of the argument relative to the claim, and the regression head maps this embedding to a base score.

Then, the BSM is trained in two phases. In the first phase (① in Figure 2) the regression head is primed with no regard for the prediction task. Thus, the parameters of $f_{\phi_{llm}}$ are frozen and only the parameters of $f_{\phi_{reg}}$ are allowed to update. In this phase, claim, argument and base score triples (a_c, a, t) are used in a supervised manner to update the regression head. In all our experiments, the target base scores t are generated by directly prompting the base LLM, optimising the regression head to use argument quality information already reflected in the model embedding. For each claim-argument pair, we compute $\hat{t} = f_\phi(a_c, a, \mathbf{1}_{a \in S_c})$. Using a standard gradient-based optimiser, we can update the parameters ϕ_{reg} using a mean square error loss function

$$\mathcal{L}_{primed}(\hat{t}, t) = \mathcal{L}_{MSE}(\hat{t}, t).$$

This step is shown in Figure 2 as the priming loss.

In the second phase of training (② in Figure 2), all trainable model parameters can freely update. In this phase, the BSM is trained using weak supervision for the claim verification task. Each step of training requires a provided claim, sets of supporting and attacking arguments, and the ground-truth labelling of the claim. For each argument a base score is computed with f_ϕ followed by application of the argumentation semantics to give a strength for the claim. As DF-QuAD is differentiable, we can back-propagate through it to update the parameters of the BSM. The loss function is

$$\mathcal{L}_{sem}(\hat{v}_c, v_c) = \mathcal{L}_{MSE}(\hat{v}_c, v_c)$$

where $\hat{v}_c = \sigma(a_c)$ and $v_c \in \{0, 0.5, 1\}$ is a mapping of the ground truth label of the claim: *True* to 1, *False* to 0, and *Uncertain* to 0.5. This step is shown in Figure 2 as the base score loss. Note that

[3] Provided in Appendix A.2.

this is a neurosymbolic semantic loss function using argumentation constraints rather than logical constraints to weakly supervise the BSM.

Additional loss terms can be used in combination with this semantic loss to optimise for desired behaviours of the BSM. We propose two such behaviours: (i) consistency in scoring semantically similar arguments and, (ii) making the BSM-induced ranking of arguments conform to some provided ordering. For (i), the consistency loss is

$$\mathcal{L}_{con}(a_c, \mathcal{C}) = \frac{1}{n}\sum_{i=1}^{n}(t_i - \bar{t})^2$$

where $\mathcal{C} = (a_1, a_2, \ldots, a_n)$ is a tuple of arguments generated for a_c assumed to be semantically equivalent (i.e. paraphrases), $t_i = f_\phi(a_c, a_i, \mathbf{1}_{a_i \in S_c})$ and $\bar{t} = \frac{1}{n}\sum t_i$. In effect, the goal of this loss is to minimise the variance of computed base scores for arguments that are semantically similar. However, $\mathcal{C}$ must be pre-generated for all input claims, which may result in a large number of samples not feasible for training. Thus, we generate $\mathcal{C}$ for only a subset of claims and then randomly sample claim-tuple pairs therefrom and compute the associated loss. For (ii), the ranking loss is

$$\mathcal{L}_{rank}(a_c, \mathcal{P}) = \frac{2}{n \times (n-1)}\sum_{i=1}^{n}\sum_{j=i+1}^{n}\max(0, m - (t_i - t_j))$$

where $\mathcal{P} = (a_1, a_2, \ldots, a_n)$ is a tuple of arguments generated for a_c in ranked order where the base score of a_i is expected to be greater than a_j for $i < j$ and, as before, $t_i = f_\phi(a_c, a_i, \mathbf{1}_{a_i \in S_c})$. In effect, this is a pairwise margin ranking loss that encourages the base scores to conform to a predefined ordering and to be separated by at least a margin of m, which is a hyperparameter. Again, as $\mathcal{P}$ must be pre-generated, we randomly sample claim-tuple pairs to compute this loss.

The full loss in phase 2 can be summarised as:

$$\mathcal{L} = \mathcal{L}_{sem} + \lambda_{rank}\mathcal{L}_{rank} + \lambda_{con}\mathcal{L}_{con}{}^{4}$$

where λ_{rank} and λ_{con} are hyperparameters that weigh their respective loss functions.

6 Reinforcement Learning for Argument Generation

The argument generator is trained using a GRPO procedure with the aim of encouraging the generation of arguments helpful for the downstream claim verification task. The training process takes a training dataset $\mathcal{D} = \{(c_i, y_i)\}_{i=1}^{N}$ composed of claims c_i and corresponding ground-truth labels y_i, an *initial argument generator* $D_{\theta_{\text{init}}}$, a fixed BSM f_ϕ and an argumentation semantics σ. During training, each claim in the dataset is passed to the *current argument generator* $D_{\theta_{\text{curr}}}$ to generate the associated sets of supporting and attacking arguments. These arguments are then scored by the BSM, organised with their respective claim into a QBAF and evaluated by σ, yielding final strengths for the claims, as described in Section 4. For each claim, we sample G outputs from $D_{\theta_{\text{curr}}}$ to obtain the group of samples required by GRPO. For each obtained claim strength $\hat{v}_c$, we compute a scalar reward $r_{\alpha,\beta}(y, \hat{v}_c)$, with $(c, y) \in \mathcal{D}$, using our custom *reward function*, which is defined, for $0 < \alpha < \beta < 1$, as:

$$r_{\alpha,\beta}(\textit{True}, \hat{v}_c) = \begin{cases} 100 & \text{if } \hat{v}_c \geq \beta \\ 200 \times \frac{\hat{v}_c}{\beta} - 100 & \text{otherwise} \end{cases}$$

$$r_{\alpha,\beta}(\textit{False}, \hat{v}_c) = \begin{cases} 100 & \text{if } \hat{v}_c \leq \alpha \\ 200 \times \frac{1-\hat{v}_c}{1-\alpha} - 100 & \text{otherwise} \end{cases}$$

$$r_{\alpha,\beta}(\textit{Uncertain}, \hat{v}_c) = \begin{cases} 100 & \text{if } \alpha \leq \hat{v}_c \leq \beta \\ 100 - 200 \times \frac{\alpha - \hat{v}_c}{\alpha} & \text{if } \hat{v}_c < \alpha \\ 100 - 200 \times \frac{\hat{v}_c - \beta}{1-\beta} & \text{otherwise} \end{cases}$$

Note that $r_{\alpha,\beta}$ as defined is consistent with our thresholding function. This means that it provides a maximum reward for any claim strength that would result in the correct classification while assigning

[4]The three constituent losses are estimated from separate data resources. The fully specified loss is given in Appendix E.

lower rewards to scores that would result in prediction errors, decreasing linearly with the increasing distance to the corresponding threshold. Model completions that cannot be parsed into supporting and attacking arguments receive a fixed negative reward of -1000. The reward is used with the standard GRPO loss function [Shao et al., 2024] during training.

7 Data

Training uses claim-level labels together with generated supporting and attacking arguments. The semantic loss is supervised by ground-truth claim labels, while generated base scores are used for priming, and synthetic paraphrases and rankings support the optional auxiliary losses. We use separate data resources for training, validation, and final evaluation, with full construction details provided in Appendix B. Training and validation include LLM-generated arguments, while final evaluation is conducted on claims drawn from existing human-curated datasets (where necessary minimally modified to become appropriate for stand-alone claim verification). We keep the two evaluation datasets strictly separate from data used in any training resource.

For training, we construct *TFU Training Claims*, a set of 1,000 claims evenly split between labels *True*, *False*, or *Uncertain*. The *True* and *False* claims are drawn from TruthfulClaim [Freedman et al., 2025], while the *Uncertain* claims are drawn from the Rational Probabilistic Beliefs dataset [Freedman and Toni, 2025]. For training the BSMs, we generate two sets of supporting and attacking arguments for a subset of the claims in *TFU Training Claims*, using GPT-5 [Singh et al., 2025] and Qwen3-8B [Yang et al., 2025] respectively. We also generate ranking and consistency datasets for the respective losses described in Section 5. The consistency dataset is generated from a subset of the Qwen3-8B arguments based on *TFU Training Claims*, by generating three semantically-equivalent paraphrases of each argument with GPT-5. The ranking dataset is also based on a subset of *TFU Training Claims*, and uses GPT-5 to generate four candidate arguments in descending order of quality.

The validation data is composed of 150 synthetically generated claims (using GPT-5 and Claude Sonnet 4.5), evenly split across the three classes, together with supporting and attacking arguments generated by Qwen3-8B.

We evaluate on two ternary claim-verification datasets. The first, *TFU Evaluation Claims*, contains 750 claims balanced across *True*, *False*, and *Uncertain*. The *True* and *False* claims are drawn from StrategyClaim [Freedman et al., 2025], while the *Uncertain* claims are drawn from DEBATunE [Li et al., 2024a]. Where necessary, DEBATunE questions are converted into declarative factual claims using GPT-5, as well as filtered to exclude statements of opinion using simple string-matching (e.g. excluding any sample containing 'should'), and human verification. The second evaluation set contains 1,746 real-world claims from AVeriTeC [Schlichtkrull et al., 2023]. Where the claims have an ambiguous temporal component, we use GPT-5 to add the date provided in the metadata to the claim, forming a single sentence. Since the resulting AVeriTeC split is imbalanced, with 334 *True*, 1,190 *False*, and 222 *Uncertain* claims, we report balanced accuracy.

8 Experiments

We evaluate whether training the components of ITA improves ternary claim verification while preserving the constraint that predictions are computed from explicit argumentative structures. The main experiments use Qwen3-8B [Yang et al., 2025] as the underlying LLM. We use Qwen3-8B because it is a strong open-weight model that fits our compute budget and is supported by our fine-tuning stack, allowing all direct and argumentative variants to be compared under the same base model. Training is performed using low-rank adaptation (LoRA) parameter-efficient fine-tuning where applicable [Hu et al., 2022]. All argumentative variants (baselines and ours) in the main comparison use fixed thresholds at $\alpha = 1/3$ and $\beta = 2/3$ and DF-QuAD semantics. Furthermore, the argumentative baselines and trained BSM experiments in the main comparison all use the same argumentative structures generated by an untuned Qwen3-8B model, to make sure the results are only measuring variation in base score attribution (but naturally in the experiments with trained argument generators we do not impose this restriction).

We report accuracy on *TFU Evaluation Claims* and balanced accuracy on the AVeriTeC evaluation set. Further implementation details, including hyperparameters, are given in Appendix G.4.

8.1 Experimental Variations

We experiment with five classes of systems, as in Table 1: *Non-Argumentative Baselines*, *Argumentative Baselines*, *Argument Generator Model*, *Base Score Models* and *Combination of Models*.

Non-Argumentative Baselines. The first variety of baselines classifies claims without constructing QBAFs. These include the untuned Qwen3-8B model, a supervised fine-tuned direct predictor (SFT), and a direct GRPO model trained with reward on the final label. These baselines test whether ITA provides benefits beyond standard prompting, supervised fine-tuning, or reinforcement learning on labels. Full details of the baselines can be found in Appendix C.

Argumentative Baselines. The second variety of baselines constructs QBAFs but does not train ITA components. In these variants, arguments are generated by Qwen3-8B and base scores are either fixed to 0.5 or assigned by prompting Qwen3-8B. These baselines isolate the effect of formal argumentative inference without training, as in ArgLLMs [Freedman et al., 2025].

Base Score Models. These variants keep the evaluation-time arguments fixed to the same Qwen3-8B generations as in *Argumentative Baselines*, but replace constant or prompted base scores with a learned base score model. This isolates the effect of training argument scores through the downstream argumentative verdict. We consider BSMs trained on Qwen3-8B-generated arguments (BSM-Q) and GPT-5-generated arguments (BSM-G), using either the semantic loss alone or the semantic loss with ranking and consistency auxiliaries, as described in Section 5.

Argument Generator Model. Here, the argument generator is optimised with GRPO using argumentation-based feedback (see Section 6), while base scores are assigned by the prompted untuned Qwen3-8B scorer. This isolates whether reinforcement learning improves the generated argumentative structure independently of the learned base score module.

Combination of Models. These variants pair the GRPO-trained argument generator with a learned BSM. We select one BSM-Q and one BSM-G variant based on validation performance, testing whether trained generation and trained scoring provide complementary gains. The BSM is trained first and remains fixed during GRPO training of the argument generator.

8.2 Main Performance Results

Table 1 reports the main results. The non-argumentative (direct-prediction) baselines are strong, with supervised fine-tuning performing best on TFU and direct GRPO performing best on AVeriTeC. The overall strong performance is expected as these models are optimised to predict labels directly, without requiring the prediction to be mediated by generated arguments and formal semantics.

The training-free argumentative baselines perform below the direct baselines. Constant base scores test whether the generated graph structure alone is sufficient, while prompted Qwen scores test whether an untuned LLM can assign useful intrinsic strengths.

The learned BSMs provide the clearest gains among the argumentative variants. On TFU, the best learned scorer improves over both argumentative baselines and the strongest direct-prediction baseline. On AVeriTeC, learned base scores substantially improve over constant and prompted argumentative scoring, and approach the strongest direct baseline. In particular, the best-performing BSM on the validation data, 'BSM-G (primed & sem)', significantly outperforms argumentative baselines on both test datasets, as indicated by the confidence intervals in Table 4. These results support the central claim that formal argumentation semantics can be used as a useful training signal for argument scoring, rather than only as an inference-time aggregation rule.

The priming-only results in Table 4 further isolate the contribution of semantic training. For BSM-G, adding the semantic loss improves TFU accuracy from 0.588 to 0.677 and AVeriTeC balanced accuracy from 0.410 to 0.460. The auxiliary ranking and consistency losses improve the local properties examined in Table 2, but their effects on downstream accuracy vary across datasets and training arguments.

The GRPO-trained argument generator gives mixed but informative results. When paired with prompted Qwen3-8B scores, it improves slightly over the corresponding Qwen3-8B-generated argumentative baseline on both datasets, suggesting that argumentation-based feedback can shape generation. However, the current results indicate that generator training is less consistently beneficial

Table 1: Main performance results (accuracy on TFU and balanced accuracy on AVeriTeC). For the argumentative baselines and our models, A+B indicates which method A was used for argument generation (args) and which method B for generation of base scores. 'BSM-Q' stands for 'base score model trained on arguments generated by Qwen3-8B', and 'BSM-G' stands for 'base score model trained on arguments generated by GPT-5'. 'sem', 'rank' and 'con' refer to different losses (see Section 5). The best results overall are bolded while the best argumentative method results are underlined. See Table 4 in the SM for further results and confidence intervals.

Variant	TFU	AVeriTeC
Non-Argumentative Baselines		
Qwen direct baseline	0.635	0.454
Qwen direct SFT	0.659	0.446
Qwen direct GRPO	0.592	**0.485**
Argumentative Baselines		
Qwen args + constant 0.5 base scores	0.555	0.405
Qwen args + Qwen base scores	0.585	0.406
Argument Generator Model (Ours)		
Qwen GRPO args + Qwen base scores	0.591	0.426
Base Score Models (Ours)		
Qwen args + BSM-Q (primed & sem)	0.588	0.415
Qwen args + BSM-Q (primed & sem & rank & con)	0.590	<u>0.471</u>
Qwen args + BSM-G (primed & sem)	**<u>0.677</u>**	0.460
Qwen args + BSM-G (primed & sem & rank & con)	0.584	0.466
Combination of Models (Ours)		
Qwen GRPO args + BSM-Q (primed & sem & rank & con)	0.581	0.468
Qwen GRPO args + BSM-G (primed & sem)	0.668	0.458

than base score training. This suggests that argumentation-based reward can shape generation in a useful direction, but the effect is modest compared with base score learning.

Finally, combining GRPO training for argument generation with the best learned BSMs does not yield additive improvements. The combined systems remain competitive with the strongest argumentative models, but slightly underperform the corresponding BSMs evaluated on the original Qwen3-8B-generated arguments. This suggests that improving argument generation and improving argument scoring are not independent plug-in gains. One possible explanation is a mismatch between the arguments used to train the fixed scorer and those produced by the GRPO-trained generator. Under the current sequential pipeline, base score learning is thus the more reliable component. Future work should investigate joint or alternating optimisation of generator and scorer.

8.3 Additional BSM Experiments

We conducted further experiments to measure whether BSM training can impart desirable argumentative properties beyond downstream accuracy: consistency in scores assigned to equivalent arguments and correct ranking of arguments gradually decreasing in quality. The results are summarised in Table 2 and discussed in Appendix F. Since no unique gold intrinsic strength is assumed, these diagnostics provide a functional assessment of the learned base scores.

Further experiments examine threshold selection (Appendix G.1), alternative argumentation semantics (Appendix G.2) and a second LLM backbone (Appendix G.3).

9 Conclusion

We introduced *inference-time argumentation* (ITA), a trainable neurosymbolic framework for three-valued claim verification extending the role of argumentation from an inference-time explanation

Table 2: Diagnostic results for base score estimation strategies. Inconsistency reports the variance in scores assigned to semantically equivalent arguments (lower is better). Ranking reports Spearman rank agreement with the expected ordering of strengths (higher is better). Best results are in bold.

Variant	Inconsistency (↓)	Ranking (↑)
Argumentative Baseline		
Qwen args + Qwen base scores	0.013	0.878
Base Score Models (Ours)		
Qwen args + BSM-Q (primed & sem)	0.007	0.525
Qwen args + BSM-Q (primed & sem & rank & con)	**0.005**	**0.944**
Qwen args + BSM-G (primed & sem)	0.008	0.709
Qwen args + BSM-G (primed & sem & rank & con)	0.006	0.921

layer into a learning interface for LLM-based claim verification. Our experiments show that training the argumentative components, especially the base score model, can benefit downstream performance. Learned base score models improve over argumentative baselines, and auxiliary consistency and ranking objectives improve desirable local properties of the scores. The resulting explanations are faithful by construction, as the final verdict is derived from the generated debate, the learned base scores, and the chosen argumentation semantics, rather than justified by a post-hoc reasoning trace.

Future Work. ITA currently reasons over generated arguments without requiring them to be grounded in retrieved evidence. For practical claim verification, future work should integrate evidence retrieval so that generated support and attacks are tied to external sources rather than only to model knowledge, potentially drawing on PACAR [Zhao et al., 2024a] and LoCal [Ma et al., 2025]. Another important direction is to move beyond the single layer QBAFs used here, allowing arguments to support or attack one another and thereby represent richer debates. Future work could also conduct human-centred evaluations of the contestability offered by ITA, in the spirit of Freedman et al. [2025]. Such evaluations could assess preferences for arguments generated before and after GRPO, agreement between BSM scores and human judgments, and whether ITA explanations aid human decision-making. Finally, beyond the targeted threshold studies reported here, we leave a broader exploration of ITA's hyperparameters to future work.

Acknowledgments and Disclosure of Funding

Freedman was funded by UKRI through the CDT in Safe and Trusted Artificial Intelligence (Grant No. EP/S023356/1). Dejl, Chen and Toni were partially supported by ERC under the EU's Horizon 2020 research and innovation programme (grant agreement No. 101020934, ADIX). Toni was also funded by EPSRC (grant UKRI3928, NeSyDebates). Gould was supported by UK Research and Innovation [UKRI Centre for Doctoral Training in AI for Healthcare grant number EP/S023283/1]. Finally, Mansi was supported by the Imperial College Departmental Scholarship. All authors are grateful to the other members of the CLArg group at Imperial for useful discussions and feedback. There are no competing interests to declare.

References

Carlos Alvarez, Maxwell Bennett, and Lucy Wang. Zero-shot scientific claim verification using LLMs and citation text. In Tirthankar Ghosal, Amanpreet Singh, Anita Waard, Philipp Mayr, Aakanksha Naik, Orion Weller, Yoonjoo Lee, Shannon Shen, and Yanxia Qin, editors, *Proceedings of the Fourth Workshop on Scholarly Document Processing (SDP 2024)*, pages 269–276, Bangkok, Thailand, August 2024. Association for Computational Linguistics. URL `https://aclanthology.org/2024.sdp-1.25/`.

Alfonso Amayuelas, Kyle Wong, Liangming Pan, Wenhu Chen, and William Yang Wang. Knowledge of knowledge: Exploring known-unknowns uncertainty with large language models. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 6416–6432, 2024.

Katie Atkinson, Pietro Baroni, Massimiliano Giacomin, Anthony Hunter, Henry Prakken, Chris Reed, Guillermo Simari, Matthias Thimm, and Serena Villata. Toward artificial argumentation. *AI Mag.*, 38(3):25–36, September 2017. ISSN 0738-4602. doi: 10.1609/aimag.v38i3.2704. URL https://doi.org/10.1609/aimag.v38i3.2704.

Pietro Baroni, Antonio Rago, and Francesca Toni. From fine-grained properties to broad principles for gradual argumentation: A principled spectrum. *International Journal of Approximate Reasoning*, 105:252–286, 2019. ISSN 0888-613X. doi: https://doi.org/10.1016/j.ijar.2018.11.019. URL https://www.sciencedirect.com/science/article/pii/S0888613X18304651.

Yanda Chen, Joe Benton, Ansh Radhakrishnan, Jonathan Uesato, Carson Denison, John Schulman, Arushi Somani, Peter Hase, Misha Wagner, Fabien Roger, Vlad Mikulik, Samuel R. Bowman, Jan Leike, Jared Kaplan, and Ethan Perez. Reasoning models don't always say what they think, 2025. URL https://arxiv.org/abs/2505.05410.

Roi Cohen, Konstantin Dobler, Eden Biran, and Gerard de Melo. I don't know: Explicit modeling of uncertainty with an [IDK] token. *Advances in Neural Information Processing Systems*, 37: 10935–10958, 2024. URL https://www.proceedings.com/079017-0349.html.

Chrisanna Cornish and Anna Rogers. Examining the faithfulness of deepseek R1's chain-of-thought reasoning. In Aman Sinha, Raúl Vázquez, Timothee Mickus, Rohit Agarwal, Ioana Buhnila, Patrícia Schmidtová, Federica Gamba, Dilip K. Prasad, and Jörg Tiedemann, editors, *Proceedings of the 1st Workshop on Confabulation, Hallucinations and Overgeneration in Multilingual and Practical Settings (CHOMPS 2025)*, pages 11–19, Mumbai, India, December 2025. Association for Computational Linguistics. ISBN 979-8-89176-308-1. doi: 10.18653/v1/2025.chomps-main.2. URL https://aclanthology.org/2025.chomps-main.2/.

Adam Dejl, Matthew Williams, and Francesca Toni. Argumentation for explainable and globally contestable decision support with LLMs. *CoRR*, abs/2603.14643, 2026. doi: 10.48550/ARXIV.2603.14643. URL https://doi.org/10.48550/arXiv.2603.14643.

Alphaeus Dmonte, Roland Oruche, Marcos Zampieri, Prasad Calyam, and Isabelle Augenstein. Claim verification in the age of large language models: A survey. *CoRR*, abs/2408.14317, 2024. doi: 10.48550/ARXIV.2408.14317. URL https://doi.org/10.48550/arXiv.2408.14317.

Gabriel Freedman and Francesca Toni. Exploring the potential for large language models to demonstrate rational probabilistic beliefs. In *The International FLAIRS Conference Proceedings*, 2025.

Gabriel Freedman, Adam Dejl, Deniz Gorur, Xiang Yin, Antonio Rago, and Francesca Toni. Argumentative large language models for explainable and contestable claim verification. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence and Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence and Fifteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'25/IAAI'25/EAAI'25. AAAI Press, 2025. ISBN 978-1-57735-897-8. doi: 10.1609/aaai.v39i14.33637. URL https://doi.org/10.1609/aaai.v39i14.33637.

Yang Gao and Francesca Toni. Argumentation accelerated reinforcement learning for cooperative multi-agent systems. In Torsten Schaub, Gerhard Friedrich, and Barry O'Sullivan, editors, *ECAI 2014 - 21st European Conference on Artificial Intelligence, 18-22 August 2014, Prague, Czech Republic - Including Prestigious Applications of Intelligent Systems (PAIS 2014)*, Frontiers in Artificial Intelligence and Applications, pages 333–338. IOS Press, 2014. doi: 10.3233/978-1-61499-419-0-333. URL https://doi.org/10.3233/978-1-61499-419-0-333.

Adam Gould and Francesca Toni. Neuro-argumentative learning with case-based reasoning. In Leilani H. Gilpin, Eleonora Giunchiglia, Pascal Hitzler, and Emile van Krieken, editors, *Proceedings of The 19th International Conference on Neurosymbolic Learning and Reasoning (NeSy 2025), 8-10 September 2025, Santa Cruz, CA, USA*, Proceedings of Machine Learning Research, pages 1090–1106. PMLR, 2025. URL https://proceedings.mlr.press/v284/gould25a.html.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, September 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Sanyam Kapoor, Nate Gruver, Manley Roberts, Katherine M Collins, Arka Pal, Umang Bhatt, Adrian Weller, Samuel Dooley, Micah Goldblum, and Andrew Gordon Wilson. Large language models must be taught to know what they don't know. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Ming Li, Jiuhai Chen, Lichang Chen, and Tianyi Zhou. Can LLMs speak for diverse people? Tuning LLMs via debate to generate controllable controversial statements. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics ACL 2024*, pages 16160–16176, Bangkok, Thailand and virtual meeting, August 2024a. Association for Computational Linguistics. URL https://aclanthology.org/2024.findings-acl.956.

Moxin Li, Yong Zhao, Yang Deng, Wenxuan Zhang, Shuaiyi Li, Wenya Xie, See-Kiong Ng, and Tat-Seng Chua. Knowledge boundary of large language models: A survey. *arXiv preprint arXiv:2412.12472*, 2024b.

Jiatong Ma, Linmei Hu, Rang Li, and Wenbo Fu. Local: Logical and causal fact-checking with llm-based multi-agents. In *Proceedings of the ACM on Web Conference 2025*, WWW '25, page 1614–1625, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400712746. doi: 10.1145/3696410.3714748. URL https://doi.org/10.1145/3696410.3714748.

Nico Potyka. Continuous dynamical systems for weighted bipolar argumentation. In *International Conference on Principles of Knowledge Representation and Reasoning*, 2018. URL https://api.semanticscholar.org/CorpusID:53015748.

Maurizio Proietti and Francesca Toni. A roadmap for neuro-argumentative learning. In Artur S. d'Avila Garcez, Tarek R. Besold, Marco Gori, and Ernesto Jiménez-Ruiz, editors, *Proceedings of the 17th International Workshop on Neural-Symbolic Learning and Reasoning, La Certosa di Pontignano, Siena, Italy, July 3-5, 2023*, CEUR Workshop Proceedings, pages 1–8. CEUR-WS.org, 2023. URL https://ceur-ws.org/Vol-3432/paper1.pdf.

Antonio Rago, Francesca Toni, Marco Aurisicchio, and Pietro Baroni. Discontinuity-free decision support with quantitative argumentation debates. In *International Conference on Principles of Knowledge Representation and Reasoning*, 2016. URL https://api.semanticscholar.org/CorpusID:32233959.

Antonio Rago, Kristijonas Čyras, Jack Mumford, and Oana Cocarascu. Argumentation and machine learning. In Dov Gabbay, Gabriele Kern-Isberner, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 3. College Publications, 2024.

Michael Schlichtkrull, Zhijiang Guo, and Andreas Vlachos. AVERITEC: a dataset for real-world claim verification with evidence from the web. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. OpenAI GPT-5 system card. *arXiv preprint arXiv:2601.03267*, 2025.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1074. URL https://aclanthology.org/N18-1074/.

Petros Vasileiadis, Emanuele De Angelis, Maurizio Proietti, and Francesca Toni. Neuro-argumentative learning with legal text. In *Proceedings of the 1st International Workshop on Advanced Neuro-Symbolic Applications (ANSyA 2025), co-located with the 28th European Conference on Artificial Intelligence (ECAI 2025)*, volume 4125. CEUR Workshop Proceedings, 2025. URL https://ceur-ws.org/Vol-4125/paper_22.pdf.

Juraj Vladika, Ivana Hacajová, and Florian Matthes. Step-by-step fact verification system for medical claims with explainable reasoning. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2025 - Volume 2: Short Papers, Albuquerque, New Mexico, April 29 - May 4, 2025*, pages 805–816. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.NAACL-SHORT.68. URL https://doi.org/10.18653/v1/2025.naacl-short.68.

Francis Rhys Ward, Francesco Belardinelli, and Francesca Toni. Argumentative reward learning: Reasoning about human preferences. In *ICML 2022 Workshop on Human-Machine Collaboration and Teaming*, 2022.

Bingbing Wen, Jihan Yao, Shangbin Feng, Chenjun Xu, Yulia Tsvetkov, Bill Howe, and Lucy Lu Wang. Know your limits: A survey of abstention in large language models. *Transactions of the Association for Computational Linguistics*, 13:529–556, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Qinan Yu, Alexa Tartaglini, Peter Hase, Carlos Guestrin, and Christopher Potts. Outcome rewards do not guarantee verifiable or causally important reasoning, 2026. URL https://arxiv.org/abs/2604.22074.

Xiaoyan Zhao, Lingzhi Wang, Zhanghao Wang, Hong Cheng, Rui Zhang, and Kam-Fai Wong. PACAR: Automated fact-checking with planning and customized action reasoning using large language models. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 12564–12573, Torino, Italia, May 2024a. ELRA and ICCL. URL https://aclanthology.org/2024.lrec-main.1099/.

Yilun Zhao, Yitao Long, Tintin Jiang, Chengye Wang, Weiyuan Chen, Hongjun Liu, Xiangru Tang, Yiming Zhang, Chen Zhao, and Arman Cohan. FinDVer: Explainable claim verification over long and hybrid-content financial documents. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14739–14752, 2024b.

Kevin Zhou, Adam Dejl, Gabriel Freedman, Lihu Chen, Antonio Rago, and Francesca Toni. Evaluating uncertainty quantification methods in argumentative large language models. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Findings of the Association for Computational Linguistics: EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 21700–21711. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.findings-emnlp.1184/.

Yuqicheng Zhu, Nico Potyka, Daniel Hernández, Yuan He, Zifeng Ding, Bo Xiong, Dongzhuoran Zhou, Evgeny Kharlamov, and Steffen Staab. ArgRAG: Explainable retrieval augmented generation using quantitative bipolar argumentation. In Leilani H. Gilpin, Eleonora Giunchiglia, Pascal Hitzler, and Emile van Krieken, editors, *Proceedings of The 19th International Conference on Neurosymbolic Learning and Reasoning (NeSy 2025), 8-10 September 2025, Santa Cruz, CA, USA*, Proceedings of Machine Learning Research, pages 697–718. PMLR, 2025. URL https://proceedings.mlr.press/v284/zhu25a.html.

A Prompts

The prompts shown below are used for all variations of the respective components.

A.1 Argument Generation

```
Please provide a set of short arguments supporting and attacking the
following claim.
Construct the arguments so they refer to the truthfulness of the
claim.
The arguments should be short and concise, ideally one sentence per
argument.
The arguments should be based on factual information and logical
reasoning.
Only provide supporting or attacking arguments if you think there
is a valid and convincing support or attack for this claim, meaning
there is a non-zero probability that the claim is true or false;
otherwise return: N/A.
Provide as many unique supporting and attacking arguments as you
think are correct and applicable to the claim.


Respond in the following format:


claim:
    ‘‘{claim}’’.


Output:
{
    'support': [
        ‘‘<SUPPORT ARGUMENT 1>’’,
        ‘‘<SUPPORT ARGUMENT 2>’’,
        ...
    ],
    'attack': [
        ‘‘<ATTACK ARGUMENT 1>’’,
        ‘‘<ATTACK ARGUMENT 2>’’,
        ...
    ]
}


Claim: {claim}
```

A.2 Base Score Model

```
You are an analyst evaluating the validity and relevance of
arguments.


For the argument:


Argument: ‘‘{statement}’’


Please give your confidence that the argument presents a compelling
case {in favour of / against} the statement:


Statement: ‘‘{claim}’’


Your assessment should be based on how well the argument {supports
/ refutes} the considered statement, as well as the correctness,
```

```
accuracy, and truthfulness of the given argument.
Your response should be between 0% and 100%, with 0% indicating that
the considered argument is definitely invalid, 100% indicating that
the considered argument is definitely valid, and values in between
indicating various levels of uncertainty.
Your estimates should be well-calibrated, so feel free to err on the
side of caution and output moderate probabilities if you are not
completely sure in your assessment.
Please respond in the following form:


[The predicted likelihood that the considered argument is valid as a
number between 0 and 100]%


Reply only with the predicted likelihood without any additional text
or explanation.
```

A.3 Baselines

```
You are a careful fact-checking assistant.
Classify the claim below using exactly one of the following three
choices:  True, False, or Uncertain.
Return only your choice as a single word and nothing else.

Claim:  {claim}
```

B Datasets

We use separate data resources for claim labels, argument generation, base score training, and final evaluation. Training and validation partially consist of synthetic arguments produced by LLMs, while final evaluation is conducted on claims drawn from existing human-curated datasets. No evaluation claims are used to construct the training resources.

B.1 Training

TFU Training Claims. We construct a training set of 1,000 claims with labels in $\{True, False, Uncertain\}$. The set contains 333 *True* claims, 333 *False* claims, and 334 *Uncertain* claims. The *True* and *False* claims are drawn from TruthfulClaim [Freedman et al., 2025], while the *Uncertain* claims are drawn from the Rational Probabilistic Beliefs dataset [Freedman and Toni, 2025].

TFU Training Arguments. For 1000 claims from TFU Training Claims, we generate supporting and attacking arguments using Qwen3-8B. Separately, we also generate arguments for 499 of these claims with GPT-5. These claim-argument sets are used to train the base score model components.

Consistency Training Arguments. We construct 2,330 paraphrase groups covering 284 claims from *TFU Training Arguments*. Each group contains three semantically equivalent rewordings, generated by GPT-5, of a single argument. These groups are used to train the base score model to assign consistent scores to semantically equivalent arguments.

Ranking Training Arguments. We construct ranking data for 493 claims from *TFU Training Claims*. Each instance contains a list of four arguments ranked from strongest to weakest with respect to a fixed claim and relation type, giving 164 supporting sets and 329 attacking sets. Arguments are generated with GPT-5 and are then filtered by twice providing them as input to Claude Sonnet 4.5 in a random order, and removing any that are not returned in the 'correct' generated order on both occasions. These data are used to train the base score module to respect relative argument strength.

Table 3: Summary of data resources used for training, validation, and evaluation.

Resource	Size	Provenance	Use
TFU Training Claims	1,000	TruthfulClaim, RPB	Claim-label training
TFU Training Arguments	499/1000 sets	GPT-5, Qwen3-8B	Argumentative training
Consistency Training	2,330 groups	GPT-5	base score consistency
Ranking Training	493 groups	GPT-5, Claude Sonnet 4.5	base score ranking
TFU Validation Claims	150	GPT-5, Claude Sonnet 4.5	Validation
TFU Evaluation Claims	750	StrategyClaim, DEBATunE	Final evaluation
AVeriTeC Claims	1,746	AVeriTeC	Final evaluation

B.2 Validation

TFU Validation Claims. We use 150 validation claims generated synthetically by Claude Sonnet 4.5 and GPT-5, evenly split across *True*, *False*, and *Uncertain*.

TFU Validation Arguments. For each validation claim, we generate supporting and attacking arguments using Qwen3-8B, yielding 150 validation claim-argument sets.

B.3 Evaluation

Final evaluation is conducted on claims drawn from existing human-curated datasets. Where necessary, we use LLM-assisted rewriting only to convert source items into declarative factual claims.

TFU Evaluation Claims. We construct a balanced evaluation set of 750 claims, with 249, 251, and 250 *True*, *False*, and *Uncertain* claims, respectively. The *True* and *False* claims are drawn from StrategyClaim [Freedman et al., 2025], while the *Uncertain* claims are drawn from DEBATunE [Li et al., 2024a]. Where necessary, questions from DEBATunE are transformed into declarative factual claims using GPT-5 and filtered to exclude opinion statements. For example, "The use of robotics in manufacturing leads to better product quality." is retained as a factual claim, whereas statements involving prescriptions such as "We should ban lotteries." are excluded.

TFU Evaluation Arguments. For each TFU Evaluation Claim, we generate supporting and attacking arguments using Qwen3-8B, yielding 750 claim-argument sets. These are used to evaluate the base score model on arguments generated by the same model family used at test time.

AVeriTeC Claims. We also evaluate on 1,746 real-world claims drawn from AVeriTeC [Schlichtkrull et al., 2023]. The resulting three-valued set is class imbalanced, with 1,190 *False*, 334 *True*, and 222 *Uncertain* claims, so we report balanced accuracy. For time-sensitive claims, we use the metadata accompanying the dataset to make the temporal context explicit without adding external information. For example, "All government schools in India are being privatised." is rewritten as "As of August 22, 2020, all government schools in India are being privatised.". To assign labels, we map SUPPORTED to *True*, REFUTED to *False*, and CONFLICTING EVIDENCE/CHERRY-PICKING to *Uncertain*. NOT ENOUGH EVIDENCE claims are disregarded, as we are more interested in claims that are uncertain due to conflicting information rather than just a lack of evidence.

AVeriTeC Arguments. For each AVeriTeC claim, we generate supporting and attacking arguments using Qwen3-8B, yielding 1,746 claim-argument sets.

Consistency Evaluation Arguments. We generate 150 sets of paraphrases for evaluation, using the same methodology as *Consistency Training Arguments* in Appendix B.1, but instead using *TFU Evaluation Arguments*.

Ranking Evaluation Arguments. Likewise, we construct an additional 149 sets of ranking arguments using the same methodology as *Ranking Training Arguments*, but based on *TFU Evaluation Claims*.

C Direct Prediction Baselines

We compare our argumentative models against direct prediction baselines that use the same underlying LLM but do not construct an argumentation framework. These baselines are intended to isolate the contribution of inference-time argumentation from the benefits of standard prompting, supervised fine-tuning, and reinforcement learning on the final label. In all direct baselines, the model is given a claim c and is prompted to output exactly one label from $\{\textit{True}, \textit{False}, \textit{Uncertain}\}$.

Qwen Direct Baseline. The first baseline is an untuned direct predictor. We prompt Qwen3-8B to classify each claim directly as *True*, *False*, or *Uncertain*. At evaluation time, generations are decoded greedily and the first valid label produced after any reasoning trace is parsed as the prediction. If no valid label can be parsed, the prediction is set to *Uncertain*. This baseline measures the performance of the base model under the same three-valued output space as our method.

Qwen Direct SFT. The second baseline fine-tunes Qwen3-8B directly on labelled claims, *TFU Training Claims* (see Appendix B.1). Given a claim, the model is trained to output only the gold label. We use parameter efficient LoRA fine-tuning, applying the language modelling loss only to the label tokens and masking the prompt tokens. This baseline tests whether the proposed argumentative pipeline improves over a model trained directly to map claims to labels using the same supervised data.

Qwen Direct GRPO. The third baseline optimises the direct predictor using GRPO with an exact-match classification reward, also using *TFU Training Claims*. For each claim, the model samples label predictions under the same direct classification prompt. A completion receives positive reward if the parsed label matches the gold label and negative reward otherwise. In our implementation, this reward is $+100$ for a correct label and -100 for an incorrect or unparseable label. This baseline controls for the possibility that any gains from our full system are due simply to reinforcement learning on the final verification label, rather than to the argumentative structure or the use of formal semantics.

D Argument Semantics

In the main experiments we use the discontinuity-free quantitative argumentation debate semantics, DF-QuAD [Rago et al., 2016]. Let

$$Q = \langle A, R^-, R^+, \tau \rangle$$

be a QBAF. For an argument $a \in A$, let $\mathrm{Att}(a) = \{b : (b, a) \in R^-\}$ and $\mathrm{Sup}(a) = \{b : (b, a) \in R^+\}$ denote its direct attackers and supporters. DF-QuAD first combines a set of argument strengths using

$$F(x_1, \ldots, x_k) = \begin{cases} 0 & \text{if } k = 0, \\ 1 - \prod_{i=1}^{k} (1 - x_i) & \text{otherwise.} \end{cases}$$

Thus multiple attackers or supporters are aggregated into a single attacking or supporting force. For an argument a, let

$$v_a^- = F\big(\{\sigma_Q(b) : b \in \mathrm{Att}(a)\}\big), \qquad v_a^+ = F\big(\{\sigma_Q(b) : b \in \mathrm{Sup}(a)\}\big),$$

where $\sigma_Q(b)$ is the final strength assigned to argument b. The final strength of a is then obtained by comparing the aggregate attacking and supporting forces:

$$\sigma_Q(a) = \begin{cases} \tau(a) & \text{if } v_a^- = v_a^+, \\ \tau(a)\big(1 - (v_a^- - v_a^+)\big) & \text{if } v_a^- > v_a^+, \\ \tau(a) + (1 - \tau(a))(v_a^+ - v_a^-) & \text{if } v_a^+ > v_a^-. \end{cases}$$

Given fixed base scores and a fixed QBAF, DF-QuAD deterministically returns the final strengths of all arguments.

E BSM Full Loss Function

The full BSM loss is defined over three data resources: the semantic training data $\mathcal{D}_{\text{sem}}$, the ranking data $\mathcal{D}_{\text{rank}}$, and the consistency data $\mathcal{D}_{\text{con}}$. It is given by

$$\begin{aligned}\mathcal{L} = \mathbb{E}_{(\hat{v}_c, v_c)\sim\mathcal{D}_{sem}}\left[\mathcal{L}_{sem}(\hat{v}_c, v_c)\right] \\ + \lambda_{rank}\mathbb{E}_{(a_c,\mathcal{P})\sim\mathcal{D}_{rank}}\left[\mathcal{L}_{rank}(a_c, \mathcal{P})\right] \\ + \lambda_{con}\mathbb{E}_{(a_c,\mathcal{C})\sim\mathcal{D}_{con}}\left[\mathcal{L}_{con}(a_c, \mathcal{C})\right].\end{aligned}$$

F Ranking and Consistency Experiments

The main experiments evaluate BSMs through their effect on the downstream accuracy of the final argumentative verdict. We additionally assess whether the learned base scores satisfy two local desiderata for argument evaluation. First, scores should be stable under paraphrase: semantically equivalent arguments should receive similar base scores. Second, scores should preserve relative argument strength: arguments ranked as stronger should receive higher scores than weaker arguments. To measure this, we conduct experiments on constructed datasets, described in the *Consistency Evaluation Arguments* and *Ranking Evaluation Arguments* sections in Appendix B.3. The results are reported in Table 2. Inconsistency is measured as the variance of the scores assigned to semantically equivalent arguments, averaged across paraphrase groups, so lower values indicate greater stability. Ranking is measured as rank agreement with the expected ordering of argument strengths, so higher values indicate better sensitivity to relative strength.

Both the training regimes using just the priming and semantic loss, as well as those using the auxiliary consistency and ranking losses, substantially improve the consistency of the BSMs compared to the baseline. It is of note that even when not specifically optimising for consistency during training we observe a significant improvement. This is important, as it is the more robust of the two tasks (it is relatively simple for an LLM to generate semantically-equivalent paraphrases of an argument, compared to coming up with an accurate quality ranking), and is a fundamental property of a good base score predictor. The ranking performance worsens on the BSMs only trained on the semantic loss (possibly in part due to noise introduced by the synthetic nature of the dataset), but significantly improves on those trained on the auxiliary losses.

G Experimental Details

G.1 Threshold Selection

We repeat GRPO training with the two selected fixed BSMs and thresholds of 0.4 and 0.6. We also optimise symmetric thresholds using training data while keeping the BSM fixed. We consider threshold pairs $(0.5 - d, 0.5 + d)$, where d is a midpoint between adjacent absolute distances of training claim strengths from 0.5, and select the pair with the best training performance. For the final two configurations in Table 5, thresholds are recalibrated on generated arguments during GRPO. The effects vary across configurations and datasets, indicating that threshold selection can affect the performance of the combined system.

G.2 Alternative Argumentation Semantics

To examine the effect of the choice of argumentation semantics, we retrain and evaluate all four BSM configurations using Quadratic Energy (QE) semantics [Potyka, 2018], keeping the remaining protocol fixed. Table 6 compares these results with DF-QuAD. Performance varies across configurations and datasets, with neither semantics consistently outperforming the other. These results show that BSM training can also be carried out with QE, while the choice of semantics affects downstream performance.

G.3 Experiments with a Second Backbone

We replicate the principal BSM experiments using Gemma-4-E4B-it, with Gemma-generated evaluation arguments and the same DF-QuAD pipeline. Table 7 reports the results. The strongest

Table 4: Main performance results (accuracy on TFU and balanced accuracy on AVeriTeC) with 95% confidence intervals estimated using BCa bootstrap with 10,000 resamples. For the argumentative baselines and our models, A+B indicates which method A was used for argument generation (args) and which method B for generation of base scores. 'BSM-Q' stands for 'base score model trained on arguments generated by Qwen3-8B', and 'BSM-G' stands for 'base score model trained on arguments generated by GPT-5'. 'sem', 'rank' and 'con' refer to different losses (see Section 5). The best results overall are bolded while the best argumentative method results are underlined.

Variant	TFU	AVeriTeC
Non-Argumentative Baselines		
Qwen direct baseline	0.635 (0.599, 0.668)	0.454 (0.427, 0.483)
Qwen direct SFT	0.659 (0.624, 0.692)	0.446 (0.421, 0.472)
Qwen direct GRPO	0.592 (0.556, 0.625)	**0.485 (0.461, 0.511)**
Argumentative Baselines		
Qwen args + constant 0.5 base scores	0.555 (0.519, 0.591)	0.405 (0.382, 0.427)
Qwen args + Qwen base scores	0.585 (0.549, 0.620)	0.406 (0.381, 0.430)
Argument Generator Model (Ours)		
Qwen GRPO args + Qwen base scores	0.591 (0.555, 0.625)	0.426 (0.401, 0.451)
Base Score Models (Ours)		
Qwen args + BSM-Q (primed)	0.583 (0.546, 0.618)	0.408 (0.383, 0.431)
Qwen args + BSM-Q (primed & sem)	0.588 (0.552, 0.623)	0.415 (0.388, 0.442)
Qwen args + BSM-Q (primed & sem & rank & con)	0.590 (0.553, 0.626)	0.471 (0.442, 0.501)
Qwen args + BSM-G (primed)	0.588 (0.552, 0.623)	0.410 (0.386, 0.435)
Qwen args + BSM-G (primed & sem)	**0.677 (0.643, 0.712)**	0.460 (0.431, 0.488)
Qwen args + BSM-G (primed & sem & rank & con)	0.584 (0.548, 0.619)	0.466 (0.438, 0.495)
Combination of Models (Ours)		
Qwen GRPO args + BSM-Q (primed & sem & rank & con)	0.581 (0.545, 0.616)	0.468 (0.439, 0.498)
Qwen GRPO args + BSM-G (primed & sem)	0.668 (0.635, 0.701)	0.458 (0.429, 0.487)

Table 5: Threshold experiments, reporting accuracy on TFU and balanced accuracy on AVeriTeC. The first two rows use thresholds of 0.4 and 0.6. The remaining rows use optimised symmetric thresholds, with recalibration during GRPO for the final two configurations.

Variant	TFU	AVeriTeC
Fixed thresholds at 0.4 *and* 0.6		
Qwen GRPO args + BSM-Q (primed & sem & rank & con)	0.538	0.485
Qwen GRPO args + BSM-G (primed & sem)	0.673	0.437
Optimised symmetric thresholds		
Qwen args + BSM-Q (primed & sem)	0.610	0.426
Qwen args + BSM-Q (primed & sem & rank & con)	0.560	0.475
Qwen args + BSM-G (primed & sem)	0.678	0.461
Qwen args + BSM-G (primed & sem & rank & con)	0.576	0.463
Qwen GRPO args + BSM-Q (primed & sem & rank & con)	0.578	0.435
Qwen GRPO args + BSM-G (primed & sem)	0.672	0.442

learned scorer on each dataset improves over the strongest training-free argumentative baseline by 2.5 percentage points on TFU and 3.7 percentage points on AVeriTeC. The best TFU result approaches direct prediction, while a larger gap remains on AVeriTeC. These results provide evidence that base score learning can benefit a second backbone, although the magnitude of the improvement depends on the model.

Table 6: BSMs retrained with QE, compared with the corresponding DF-QuAD configurations. We report accuracy on TFU and balanced accuracy on AVeriTeC.

	QE		DF-QuAD	
Variant	TFU	AVeriTeC	TFU	AVeriTeC
BSM-G (primed & sem)	0.681	0.445	0.677	0.460
BSM-G (primed & sem & rank & con)	0.680	0.458	0.584	0.466
BSM-Q (primed & sem)	0.568	0.464	0.588	0.415
BSM-Q (primed & sem & rank & con)	0.611	0.445	0.590	0.471

Table 7: Gemma results, reporting accuracy on TFU and balanced accuracy on AVeriTeC. BSM-Gemma and BSM-GPT denote scorers trained on arguments generated by Gemma and GPT, respectively. All argumentative variants use Gemma-generated evaluation arguments.

Variant	TFU	AVeriTeC
Gemma direct	0.524	0.444
Gemma args + constant base scores	0.404	0.370
Gemma args + Gemma base scores	0.492	0.364
Gemma args + BSM-Gemma (primed & sem)	0.517	0.404
Gemma args + BSM-Gemma (primed & sem & rank & con)	0.492	0.391
Gemma args + BSM-GPT (primed & sem)	0.491	0.370
Gemma args + BSM-GPT (primed & sem & rank & con)	0.509	0.374

G.4 Architecture and Hyperparameters

G.4.1 Direct Baselines

For the direct SFT and direct GRPO baselines, we use LoRA adapters with rank 8, $\alpha = 32$, dropout 0.05, and target the query, key, value, and output projection matrices. The SFT baseline is trained for three epochs with learning rate 2×10^{-4}, batch size 1, and gradient accumulation over 8 steps. The GRPO baseline is trained for three epochs with learning rate 10^{-5}, batch size 1, gradient accumulation over 8 steps, 4 generations per prompt, temperature 1.0, and maximum completion lengths of 512 tokens.

G.4.2 Argumentative Baselines

These are the same as in [Freedman et al., 2025], apart from being restricted to a depth of 1 and not placing a restriction on the width of the frameworks (the original had a variant with depth of 2 and only allowed a maximum of a single supporting and attacking argument in each layer). Furthermore, for this baseline, there is no variation where the claim has an estimated base score (it is always fixed at 0.5), unlike in the original where this is one of the configurations.

G.4.3 Base Score Model

During the priming stage of training, we only optimized the BSM regression head, a two-layer MLP with a hidden size of 256, a ReLU hidden activation and a sigmoid output activation. The model was optimized using an AdamW optimizer with a learning rate of 1×10^{-4}, a dropout of 0.1 and a batch size of 8. Training was run over a single epoch.

During the second training phase relying on the semantic loss and optionally the ranking and consistency loss, we optimised the regression head as well as LoRA adapters with rank 8, $\alpha = 32$ and dropout 0.1 targeting the query, key, value, output, gate, up and down projection matrices. We again used an AdamW optimizer with a learning rate 1×10^{-4}, but used gradient accumulation over 8 steps and gradient checkpointing to reduce memory requirements. In settings where both auxiliary losses were used, we set $\lambda_{\text{rank}} = 1$ and $\lambda_{\text{con}} = 1$ with $m = 0.1$. As for priming, the training was done over a single epoch, with samples for the ranking and consistency losses being randomly sampled at each main training step.

G.4.4 Reinforcement Learning for Argument Generation

We fine-tune Qwen3-8B using Group Relative Policy Optimisation (GRPO) with standard HuggingFace Transformers and PEFT. To reduce memory requirements, the model is loaded in bfloat16 precision with gradient checkpointing enabled. We attach LoRA adapters of rank $r = 8$, scaling factor $\alpha = 32$, and dropout 0.1, targeting the query, key, value and output matrices. The model is optimised using a learning rate of 1×10^{-5} over 2 epochs, with a per-device batch size of 2 and gradient accumulation over 4 steps, yielding an effective batch size of 8. Each GRPO step samples 2 completions per prompt, with a maximum prompt length of 512 tokens and a maximum completion length of $2{,}048$ tokens.

The reward is computed from the final strength of the claim after evaluating its generated supporting and attacking arguments. During GRPO, either a fixed learned BSM or a frozen prompted LLM assigns base scores to the generated arguments, while the claim's own base score remains fixed at 0.5. Only the argument generator's LoRA parameters are updated. All remaining training hyperparameters are identical across both configurations.

G.5 Computational Resources

Base Score Model We trained the BSMs on three NVIDIA A100 80GB accessed via a cloud server running a customized distribution of Ubuntu 24.04.2. On this hardware, the generation of the target base scores for priming via direct prompting took approximately 3 minutes, with the subsequent training on these scores taking approximately 2 minutes. The second phase of the training took approximately 10 minutes when not using the ranking and consistency losses and approximately 20 minutes with these losses.

Argument Generation Model Similar to the BSMs, we finetuned the argument generators on three NVIDIA A100 80GB GPUs accessed via a cloud server running a customized distribution of Ubuntu 24.04.2. On this hardware, the training for 450 steps took 11 hours using the learned BSM and 19 hours using the LLM-based BSM.

H Licences

Qwen3-8B and Gemma-4-E4B-it are released under Apache-2.0. TruthfulQA and StrategyQA, which underlie TruthfulClaim and StrategyClaim, are released under Apache-2.0 and MIT respectively. The ArgLLMs repository containing the adapted claim-verification datasets is released under the Argumentative LLMs Academic Public Licence, which permits non-commercial academic use. RPB is released under MIT, while AVeriTeC is released under CC BY-NC 4.0. DEBATunE does not appear to provide an explicit redistribution licence, however we reached out to the authors and are still awaiting a response. Other releases by the research group use a BSD 3-Clause License. Synthetic arguments and auxiliary labels generated using GPT-5 and Claude are subject to the applicable OpenAI and Anthropic terms of service.

I Limitations

This work studies claim-only verification without external evidence retrieval. Generated arguments may therefore be plausible but factually unsupported. Incorporating approaches such as RAG into our framework, in the spirit of ArgRAG [Zhu et al., 2025], is a promising avenue for future work. While ITA provides procedural faithfulness, i.e., the verdict deterministically follows from the considered argumentative structure, scores, thresholds, and semantics, the arguments themselves may be incomplete or incorrect. Our QBAFs are also restricted to depth-one structures, and some supervision signals are synthetically generated, which may introduce artifacts. Finally, we limit ourselves to the task of ternary claim verification. Future work could explore the extension of our framework to domains involving open-ended decision-making, e.g., drawing from the methodology of ArgEval [Dejl et al., 2026], an argumentative approach designed for such domains.

Due to a lack of computational resources, we were limited in the scope of experiments we could carry out. Beyond the targeted threshold studies in Appendix G.1, we were not able to carry out a broader hyperparameter search. The remaining settings were chosen by estimation and following conventions

from the literature rather than empirically validated as the best choice, and therefore our presented results may be an underestimation of the true potential performance of the proposed method.

J Safeguards

ITA should be used as an interpretable research tool, not as an automated authority on factual truth. Its outputs should be inspected against trusted evidence, especially in high-stakes domains. Generated arguments and scores should be presented as model-produced rationales, not as verified evidence.

K Broader Impacts

This work may improve transparency in claim verification by making predictions reconstructible from explicit argumentative structures. However, fluent generated arguments can also make incorrect verdicts appear more credible. Real-world use should therefore pair argumentative explanations with evidence retrieval, source attribution, and human review.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The contributions of the paper are clearly laid out in the abstract and introduction, and the rest of the paper lays out their implementation and discusses the results.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Discussed in Appendix I.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

Justification: N/A

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The method is fully described in Section 4-Section 6, with additional details on prompts in Appendix A. The experimental set-up is fully defined in Section 8 and details on hyper-parameters are given in Appendix G.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Code and datasets are linked to in the abstract.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Hyperparameters given in Appendix G.4, and code is provided as supplemental material.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: These are given in Appendix G.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Provided in Appendix G.5.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The guidelines have been reviewed and followed throughout the paper.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Discussed in Appendix K.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: Discussed in Appendix J.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Existing datasets and models have been cited (Appendix B) and licenses included where available (Appendix H).

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Documentation is included in the codebase, which is linked from the abstract.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: N/A

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: N/A

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: LLMs are used as a core component in the methodology and for baselines. These are described in detail in Section 8 and Appendix G.4.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.