Einon at CheckThat! 2026: Reward Model with Confidence-Based Filtering for Numerical Claim Verification

Task 2: Test-Time Scaling for Fact Verification

Jiacheng Tang[1], Zhongyuan Han[1,*], Jieren Luo[1] and Kaichuan Lin[1]

[1]*Foshan University, Foshan, China*

Abstract

This paper describes our participation in the CLEF 2026 CheckThat! Lab Task 2 (English track), which focuses on test-time scaling for fact verification of numerical claims. The task requires ranking multiple reasoning traces generated by language models and selecting the most reliable verdict. We propose a two-stage approach that first trains a BERT-based reward model to score reasoning quality, then applies confidence-based filtering with adaptive thresholding to select high-quality reasoning traces for final verdict prediction. Our official submission (ID: 716924) used Claude Sonnet 4.6 as a zero-shot reasoning-path ranker and achieved an average score of 0.3996. We additionally report post-deadline BERT-based ablation runs, used only for analysis, where confidence-based filtering reached 0.4260. These results clarify the contribution of thresholded verdict selection in this shared-task setting.

Keywords

Fact-checking, Numerical claims, Reward model, Test-time scaling, Reasoning trace ranking

1. Introduction

The proliferation of misinformation, particularly claims involving numerical data, poses significant challenges for automated fact-checking systems [1, 2]. The CLEF 2026 CheckThat! Lab Task 2 addresses this challenge by introducing a test-time scaling approach for fact verification [3, 4]. Unlike traditional fact-checking tasks that focus solely on classification accuracy [5, 6], this task emphasizes the quality assessment of reasoning processes generated by large language models (LLMs).

Given a numerical claim and a set of candidate reasoning traces with their corresponding verdicts (True, False, or Conflicting), the task requires systems to rank the reasoning traces by quality and select the most reliable verdict from the top-5 ranked traces. The evaluation metrics include Recall@5 for ranking quality, and Macro F1 for verdict classification accuracy. This approach is inspired by recent work on reasoning verification [7, 8] and chain-of-thought prompting [9].

Our approach combines supervised learning with statistical confidence estimation. We train a BERT-based reward model [10, 11] to score reasoning quality, then apply adaptive confidence-based filtering to identify high-quality reasoning traces for final verdict selection. This paper focuses on our official submission and uses post-deadline runs only as controlled ablations to analyze verdict-selection strategies.

2. Task Description

2.1. Problem Formulation

For each test instance, the system receives:

CLEF 2026 Working Notes, 21 – 24 September 2026, Jena, Germany

*Corresponding author.

1170909163@qq.com (J. Tang); hanzhongyuan@gmail.com (Z. Han); wolike666@gmail.com (J. Luo); 1441584221bruan@gmail.com (K. Lin)

0009-0008-7343-8982 (J. Tang); 0000-0001-8960-9872 (Z. Han); 0009-0007-4867-4214 (J. Luo); 0009-0001-7843-2888 (K. Lin)

- A numerical claim c to be verified
- A set of reasoning traces $\{r_1, r_2, \ldots, r_n\}$
- Corresponding verdicts $\{v_1, v_2, \ldots, v_n\}$ where $v_i \in \{\text{True}, \text{False}, \text{Conflicting}\}$
- Supporting evidence documents (optional)

The system must output:

- A score list $\{s_1, s_2, \ldots, s_n\}$ ranking the reasoning traces
- A final verdict v_{pred} selected from the top-5 ranked traces

2.2. Evaluation Metrics

The task employs three primary metrics:

Recall@5: Measures the proportion of correct verdicts present in the top-5 ranked reasoning traces.

MRR@5: Mean Reciprocal Rank of the first correct reasoning trace within the top-5 positions.

Macro F1: Standard macro-averaged F1-score for the predicted verdicts across three classes (True, False, Conflicting).

The final ranking is determined by the average score:

$$\text{Average Score} = \frac{\text{Macro F1} + \text{Recall@5}}{2} \tag{1}$$

3. Methodology

Our approach consists of two main components: reward model training and confidence-based verdict selection. The reward model training follows the paradigm of learning from human feedback [12, 11], while the verdict selection strategy is inspired by self-consistency methods [13].

3.1. Reward Model Training

We employ BERT-base-uncased [10] as our base architecture for the reward model. The model receives a claim, a candidate verdict, and a reasoning trace, and outputs a scalar compatibility score indicating how well that reasoning path supports the correct final label.

3.1.1. Training Data Construction

For each training instance with claim c, reasoning traces $\{r_i\}$, and ground truth label y, we construct training examples by concatenating the claim, verdict, and reasoning trace:

$$x_i = [\text{Claim: } c]\ [\text{Verdict: } v_i]\ [\text{Reasoning: } r_i] \tag{2}$$

where each component is separated by special tokens. The target label for each example is:

$$t_i = \begin{cases} 1 & \text{if } v_i = y \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

We trained on 6,400 English training claims and tuned the system on the organizer-provided validation split of 1,600 claims. For each claim, we sampled up to two reasoning traces whose verdict matched the gold label and up to four traces with mismatched verdicts. We also injected a small number of *unknown* traces (at most 150 over the full training set) to make the reward model more robust to noisy candidates. We did not apply explicit class reweighting.

3.1.2. Model Architecture and Training

The reward model uses the standard BERT-base-uncased encoder with a single regression head. In the main training script used for our best BERT-based runs, we fine-tuned the model with AdamW, a learning rate of 4×10^{-5}, batch size 64, weight decay 0.01, 500 warmup steps, and maximum sequence length 512. Training was run for 3 epochs with mixed-precision optimization, and model selection was guided by the held-out validation split.

3.2. Verdict Selection Strategies

After obtaining reward scores $\{s_1, s_2, \ldots, s_n\}$ from the trained model, we explored two main strategies for final verdict selection. The first strategy applies statistical confidence estimation to identify high-quality reasoning traces. For each test instance, we compute the mean $\mu = \frac{1}{n}\sum_{i=1}^{n} s_i$ and standard deviation $\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(s_i - \mu)^2}$ of the reward scores. We then define a confidence threshold $\theta = \mu + \alpha \cdot \sigma$, where α is a hyperparameter controlling the strictness of filtering. Higher α values result in more selective filtering.

The high-confidence reasoning traces are identified as $\mathcal{H} = \{i \mid s_i > \theta\}$. The final verdict is selected based on the size of $\mathcal{H}$: if no traces exceed the threshold, we select the verdict with the highest score; if exactly one trace exceeds the threshold, we select its verdict; if multiple traces exceed the threshold, we apply majority voting among high-confidence verdicts, breaking ties by selecting the trace with the highest score.

The second strategy, employed for the opt18 series submissions, uses a conflict-aware approach that adapts the selection mechanism based on verdict distribution. For samples without disagreement (71.4% of cases), we directly select the reasoning trace with the highest reward score. For samples with disagreement (28.6% of cases), we apply adaptive selection: if the highest score significantly exceeds others (more than $\mu + 0.4\sigma$), we select it directly; if Conflicting verdicts exist with competitive scores (score gap less than 0.15), we prioritize them; otherwise, we apply weighted voting among high-scoring traces where $s_i > \mu$. This strategy recognizes that disagreement-heavy samples require more careful handling, especially for the Conflicting class. We selected α by sweeping $\{0.2, 0.3, 0.38, 0.4, 0.5\}$ in post-deadline submissions; $\alpha = 0.4$ achieved the best score, while $\alpha = 0.38$ tied it.

4. Experiments and Results

4.1. Experimental Setup

Dataset: We used the CLEF 2026 CheckThat! Task 2 English dataset, with 6,400 training claims, 1,600 validation claims, and 2,558 test claims. The validation split contains 15 candidate reasoning traces per claim, while the final test release contains 20; our method operates on all provided traces for each instance.

Implementation: Models were implemented using PyTorch and Hugging Face Transformers library. Training was conducted on a single NVIDIA RTX 3090 GPU.

Baselines: We report two reference systems. The first is a simple majority baseline that predicts the most frequent verdict among the candidate traces without reward-model scoring. The second is an RM-only baseline that directly uses the verdict of the single highest-scoring trace. Table 1 uses the RM-only baseline because it isolates the contribution of confidence-based filtering.

4.2. Official Submission

According to the task submission guidelines, the official ranking is determined by the last submission to the leaderboard. Our official submission (ID: 716924) employed a Claude Sonnet 4.6 API-based approach for reasoning-trace ranking, achieving an average score of 0.3996. For each claim, we prompted Claude with the claim and all candidate reasoning paths, truncated each path to the first 300 characters, and

asked the model to return a JSON ranking. We used a maximum output length of 1024 tokens and otherwise kept the API settings at their defaults. The final verdict was taken from the top-ranked path. This submission successfully processed 2,540 out of 2,558 test instances (99.3%); for the remaining 18 failed cases, we fell back to the default path order. In retrospect, this official run underperformed our BERT-based variants because it used no task-specific training, relied on truncated reasoning paths, and was exposed to API failures.

4.3. Non-Official Runs

In addition to our official submission, we report earlier post-deadline runs that achieved higher performance and help explain which parts of the pipeline were most effective. Submission 703032 achieved an average score of 0.4260. Two additional submissions (704829 and 704830) using similar strategies achieved identical scores, as they employed the same final decision rule with minor parameter variations that did not affect the resulting predictions. These runs are reported only as analytical ablations and are clearly distinguished from our official ranking.

These three submissions employed different strategies:

- **703032 (opt5_strict)**: Confidence-based filtering with strict threshold ($\alpha = 0.4$)
- **704829 & 704830 (opt18 series)**: Conflict-aware strategy that applies different selection mechanisms for samples with and without verdict disagreement

4.4. Ablation Study

We conducted extensive experiments to evaluate different verdict selection strategies. Table 1 summarizes the results on the test set.

Table 1
Comparison of different verdict selection strategies

Strategy	Avg Score	Submission ID
Simple majority (no RM)	0.4073	707740
RM only (top-1)	0.4183	700194
Confidence ($\alpha = 0.2$)	0.4223	703026
Confidence ($\alpha = 0.3$)	0.4257	702950
Confidence ($\alpha = 0.38$)	0.4260	703752
Confidence ($\alpha = 0.4$)	**0.4260**	703032
Confidence ($\alpha = 0.5$)	0.4208	703262
Conflict-aware (opt18)	**0.4260**	704829, 704830

The RM-only baseline already improves over simple majority voting (0.4183 vs. 0.4073), indicating that the trained reward model provides the main gain. Confidence-based filtering adds a further +0.0077 absolute improvement over RM-only, and the conflict-aware strategy matches the best confidence-filtered score by explicitly handling disagreement-heavy cases. The best threshold range is $\alpha \in [0.38, 0.4]$; both looser and stricter filtering degrade performance. We do not report significance tests for these leaderboard deltas because the competition interface returned only aggregate scores for each submission during the competition period.

4.5. Class Distribution of the Released Test Labels

Table 2 reports the class distribution of the released English test labels. The test set is strongly imbalanced toward *False*, while *Conflicting* remains comparatively rare. This helps explain why conservative verdict-selection strategies can be competitive and why the disagreement-sensitive class is difficult to optimize reliably.

Table 2
Class distribution of the released English test labels

Label	Count	Proportion
False	1783	69.7%
True	520	20.3%
Conflicting	255	10.0%

5. Analysis and Discussion

5.1. Effectiveness of Confidence-Based Filtering

Our experiments demonstrate that confidence-based filtering provides consistent improvements over direct reward-model ranking. The key insight is that not all high-scoring reasoning traces are equally reliable. By computing statistical confidence measures and applying adaptive thresholding, we can identify a subset of traces that are more likely to contain correct verdicts.

The optimal threshold range ($\alpha \in [0.38, 0.4]$) suggests that filtering approximately the top 20-30% of reasoning traces by confidence provides the best balance between precision and recall.

5.2. Comparison with Alternative Approaches

We experimented with several alternative strategies:

Ensemble Methods: We tested homogeneous ensembles (3 BERT models with different seeds) and heterogeneous ensembles (BERT + DeBERTa + RoBERTa). However, ensemble approaches underperformed compared to single-model confidence-based filtering, likely due to increased complexity without corresponding gains in reasoning quality assessment.

Weighted Voting: We explored weighted voting schemes where votes are weighted by reward model scores. This approach showed marginal improvements but did not outperform simple majority voting among high-confidence traces.

Dynamic Top-K Selection: We implemented adaptive selection of K (the number of traces to consider) based on score distributions. This approach achieved competitive results but added complexity without substantial gains.

5.3. Performance Analysis

Our existing error logs show a strong tendency for the best BERT-based run to predict *False* (1,943 out of 2,558 cases) and comparatively few *Conflicting* labels (93 cases). Given that the released test labels contain 255 *Conflicting* instances, this under-prediction helps explain the remaining weakness on disagreement-heavy samples and motivated the conflict-aware opt18 rules. The gap between the official and post-deadline runs is also understandable in hindsight: the official system used a zero-shot API ranker with truncated reasoning paths and occasional fallback ordering, whereas the stronger runs used a task-specific reward model and a deterministic verdict-selection heuristic.

5.4. Limitations and Future Work

Our approach has several limitations:

Threshold Sensitivity: The optimal confidence threshold may vary across different claim types and domains. Future work could explore adaptive threshold selection based on claim characteristics.

Evidence Utilization: Our current approach does not explicitly incorporate the provided evidence documents. Integrating evidence-aware reasoning assessment could improve performance.

Statistical Testing: We did not perform significance testing on leaderboard scores during the competition because the platform exposed only aggregate submission metrics. A more rigorous post-hoc analysis should compare paired predictions under a fully reproducible local scorer.

Conflicting Class Handling: As suggested by our error analysis and the low frequency of Conflicting labels in the released test set, the Conflicting class remains the most challenging category. More sophisticated conflict detection mechanisms are needed to identify contradictory information across reasoning traces.

6. Conclusion

This paper presented our approach to the CLEF 2026 CheckThat! Lab Task 2 for numerical claim verification. Our method combines BERT-based reward modeling with confidence-based filtering to rank reasoning traces and select reliable verdicts.

Our official submission (ID: 716924) employed a Claude API-based approach achieving an average score of 0.3996. Post-deadline ablation runs with a task-specific BERT reward model and confidence-based filtering reached 0.4260, showing that most of the final gain came from replacing zero-shot ranking with task-specific scoring and thresholded verdict aggregation.

Future work will focus on adaptive threshold selection, evidence-aware reasoning assessment, and improved handling of conflicting information.

Acknowledgments

This work is supported by the National Social Science Foundation of China (Grant No. 24BYY080).

Declaration on Generative AI

During the preparation of this work, the authors used Claude Sonnet 4.6 for code development assistance and experimental design discussions. The authors reviewed and edited all content and take full responsibility for the publication's content.

References

[1] Z. Guo, M. Schlichtkrull, A. Vlachos, A survey on automated fact-checking, in: Transactions of the Association for Computational Linguistics, volume 10, 2022, pp. 178–206.

[2] P. Nakov, F. Alam, A. Barrón-Cedeño, G. Da San Martino, T. Elsayed, M. Hasanain, F. Haouari, B. Huet, A. Nikolov, S. Shaar, et al., Overview of checkthat! 2023: Checkworthiness, subjectivity, political bias, factuality, and authority of news articles and their source, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction, 2023, pp. 440–463.

[3] J. M. Struß, et al., Overview of checkthat! lab at clef 2026: Test-time scaling for fact verification, in: Working Notes of CLEF 2026 - Conference and Labs of the Evaluation Forum, Jena, Germany, 2026.

[4] C. Snell, J. Lee, K. Xu, A. Kumar, Scaling llm test-time compute optimally can be more effective than scaling model parameters, arXiv preprint arXiv:2408.03314 (2024).

[5] J. Thorne, A. Vlachos, C. Christodoulopoulos, A. 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), 2018, pp. 809–819.

[6] J. Chen, R. Zhong, Y. Zhu, Y. Zhang, Y. Jiang, M. Huang, Fact-checking complex claims with program-guided reasoning, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, 2022, pp. 6981–6992.

[7] H. Lightman, V. Kosaraju, Y. Burda, H. Edwards, B. Baker, T. Lee, J. Leike, J. Schulman, I. Sutskever, K. Cobbe, Let's verify step by step, arXiv preprint arXiv:2305.20050 (2023).

[8] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, et al., Training verifiers to solve math word problems, in: arXiv preprint arXiv:2110.14168, 2021.

[9] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, Chain-of-thought prompting elicits reasoning in large language models, Advances in Neural Information Processing Systems 35 (2022) 24824–24837.

[10] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019, pp. 4171–4186.

[11] P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, D. Amodei, Deep reinforcement learning from human preferences, Advances in Neural Information Processing Systems 30 (2017).

[12] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al., Training language models to follow instructions with human feedback, Advances in Neural Information Processing Systems 35 (2022) 27730–27744.

[13] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, D. Zhou, Self-consistency improves chain of thought reasoning in language models, in: International Conference on Learning Representations, 2023.