VeriFind at CheckThat! 2026: Enhancing Numerical Claim Verification via Fine-Tuned LLMs

CheckThat! Lab at CLEF 2026

Diana Zhang[1], Xiaoying Song[2,*], Mohotarema Rashid[2], Anirban Saha Anik[3], Junhua Ding[4] and Lingzi Hong[4]

[1]Texas Academy of Mathematics and Science, University of North Texas, Denton, TX, USA
[2]Department of Information Science, University of North Texas, Denton, TX, USA
[3]Department of Computer Science and Engineering, University of North Texas, Denton, TX, USA
[4]Department of Data Science, University of North Texas, Denton, TX, USA

Abstract

This paper presents our system for Task 2 of the CLEF 2026 CheckThat! Lab, which focuses on test-time scaling for numerical claim verification. The task requires ranking reasoning traces according to their usefulness in reaching the correct verdict and producing a final verdict from the top-ranked trace. We propose **TraceFT** (**Trace** Reasoning Verification with **F**ine-**T**uned LLMs), a lightweight LLM-based verifier for reasoning trace selection. TraceFT addresses this task through lightweight LLM fine-tuning: we fine-tune LLaMA-3.2-3B with Low-Rank Adaptation (LoRA) as a binary verifier that estimates whether each reasoning trace leads to the correct verdict. Experimental results show that TraceFT outperforms the official baselines, and our system **ranked 2nd** among all participating teams on the English numerical claims task, demonstrating the effectiveness of fine-tuned LLM verifiers for reasoning trace selection. Overall, our findings highlight the promise of lightweight LLM fine-tuning for numerical claim verification.

Keywords

Numerical claim verification, Fact checking, Test-time scaling, Binary verifier, Misinformation detection

1. Introduction

The growing volume of online misinformation has increased the need for automated fact verification systems, particularly for real-world claims that require evidence-based reasoning [1, 2]. Numerical and temporal claims are especially challenging because they require systems to reason over quantities, dates, statistics, comparisons, and time-sensitive evidence [3, 4]. Unlike simple factual claims, numerical claims may be partially supported by evidence while still reaching an incorrect or misleading conclusion, making them difficult for verification systems to assess reliably [4, 5]. These challenges motivate verification methods that can evaluate not only the final verdict, but also the reasoning process used to reach it.

Large language models (LLMs) have shown promise for fact verification and complex reasoning by generating step-by-step reasoning traces, often referred to as chain-of-thought reasoning [6, 7]. However, these traces are not always correct or faithful to the model's final answer, and prior work has shown that LLM-generated explanations for fact-checking can be unfaithful or unsupported by the provided evidence [8, 9]. For a given claim, an LLM may produce several plausible reasoning paths, but only some of them may accurately support the correct verdict. This observation motivates test-time selection methods, such as self-consistency and verifier-based selection, which generate multiple candidate solutions and select or aggregate the most reliable output [10, 11]. Therefore, a key challenge

CLEF 2026 Working Notes, 21 – 24 September 2026, Jena, Germany
*Corresponding author.
DianaZhang@my.unt.edu (D. Zhang); XiaoyingSong@my.unt.edu (X. Song); MohotaremaRashid@my.unt.edu (M. Rashid); AnirbanSahaAnik@my.unt.edu (A. S. Anik); Junhua.Ding@unt.edu (J. Ding); Lingzi.Hong@unt.edu (L. Hong)
0009-0006-3237-9133 (D. Zhang); 0000-0001-9390-1155 (X. Song); 0009-0003-2683-7191 (M. Rashid); 0000-0002-7824-3702 (A. S. Anik); 0000-0002-2129-1586 (J. Ding); 0000-0001-8412-8180 (L. Hong)

is not only generating candidate reasoning traces, but also ranking them according to their reliability. Recent work on process supervision and process reward models further supports this direction by showing that evaluating intermediate reasoning steps can improve the reliability of multi-step reasoning systems [12, 13].

The CLEF 2026 CheckThat! Lab is a shared evaluation effort that organizes tasks to advance multilingual fact-checking [14, 15, 16]. Task 2 of the CLEF 2026 CheckThat! Lab focuses on this problem through test-time scaling for numerical claim verification [17]. Given a claim, relevant evidence, and multiple LLM-generated reasoning traces, systems must rank the traces based on their utility in reaching the correct verdict and output the final verdict from the top-ranked trace.

To address this task, we propose **TraceFT** (**Trace** Reasoning Verification with Fine-**T**uned LLMs), a lightweight fine-tuned LLM verifier. TraceFT fine-tunes **LLaMA-3.2-3B** with **Low-Rank Adaptation (LoRA)** [18] to score each reasoning trace as likely correct or incorrect. At inference time, the model ranks all candidate traces by verifier score and selects the verdict from the highest-ranked trace. This approach treats numerical claim verification as a reasoning trace selection problem, allowing the system to prioritize trace quality over simple verdict frequency.

2. Related Work

2.1. Automated Fact Verification

Automated fact verification has become an important research direction for addressing the rapid spread of misinformation. Existing systems commonly approach the problem as a pipeline involving claim understanding, evidence retrieval, and verdict prediction [19]. Large-scale benchmarks such as FEVER formalize this setting by requiring systems to verify claims against textual evidence and classify them as supported, refuted, or not enough information [20]. Similarly, SciFact extends evidence-based verification to scientific claims, where systems must identify relevant abstracts, retrieve rationales, and predict whether the evidence supports or refutes the claim [21]. These benchmarks show that effective fact verification requires both accurate evidence interpretation and reliable reasoning over the relationship between claims and evidence.

Recent work has further explored the use of large language models for fact verification, especially in settings that require multi-step reasoning. However, LLM-generated reasoning traces are not always reliable, as they may contain unsupported assumptions, hallucinated evidence, or conclusions that do not follow from the provided evidence. This limitation is especially important for numerical and temporal claims, where small errors in quantities, dates, or comparisons can change the final verdict. Test-time scaling methods address this issue by generating multiple reasoning traces and selecting the most reliable one. For example, self-consistency improves reasoning by sampling multiple chain-of-thought paths and aggregating their answers [10]. More closely related to our work, verifier-based approaches rank candidate reasoning traces and select the trace most likely to lead to the correct verdict. Our method follows this direction by treating Task 2 as a reasoning trace verification problem rather than a direct claim classification problem.

2.2. Parameter-Efficient Fine-Tuning for LLMs

Fine-tuning large language models can improve performance on specialized downstream tasks, but full fine-tuning is often computationally expensive and memory-intensive. Parameter-efficient fine-tuning methods address this challenge by updating only a small subset of model parameters while keeping most of the pretrained model fixed. LoRA is one widely used approach that injects trainable low-rank matrices into transformer layers, reducing the number of trainable parameters while maintaining strong task adaptation performance [18]. This makes LoRA especially useful for adapting LLMs under limited computational resources.

The LLaMA family of models has shown strong general-purpose language modeling and reasoning capabilities, making it a practical foundation for downstream adaptation [22]. In our work, we fine-tune

LLaMA-3.2-3B with LoRA to create a lightweight verifier for reasoning trace quality. Rather than fine-tuning the full model, TraceFT updates only a small number of LoRA parameters, allowing efficient adaptation to the task of distinguishing correct from incorrect reasoning traces. This design enables our system to use LLM-based verification while remaining computationally feasible.

3. Task Overview

We participated in **Task 2: Fact-Checking Numerical Claims (English)** of the CLEF 2026 CheckThat! Lab [17]. The task focuses on test-time scaling for fact verification, where systems are given a claim, relevant evidence, and multiple LLM-generated reasoning traces with corresponding candidate verdicts. The goal is to rank the reasoning traces according to their utility in reaching the correct verdict and to produce the final verdict from the top-ranked trace.

Unlike earlier fact-checking settings that mainly evaluate final classification accuracy, this task explicitly incorporates reasoning trace quality into the evaluation. For each claim, multiple reasoning traces are generated by an LLM using different temperature settings to increase diversity, followed by a de-duplication step to remove redundant traces. Participants are then expected to train a verifier model that ranks these traces and selects the most reliable reasoning path for the final prediction.

Each claim is assigned one of three verdict labels:

- **True:** The provided evidence fully supports the claim.
- **False:** The provided evidence fully refutes the claim.
- **Conflicting:** The provided evidence is ambiguous, incomplete, or conflicting, and therefore neither fully supports nor fully refutes the claim.

The task is especially challenging because the claims contain quantitative quantities and/or temporal expressions. Systems must therefore reason over numerical values, dates, comparisons, and potentially incomplete or conflicting evidence.

4. Methodology

4.1. Problem Formulation

Given a claim C, its supporting evidence E, and a set of N candidate reasoning traces $T = \{t_1, t_2, \ldots, t_N\}$, the task is to identify the trace that is most useful for reaching the correct verdict. Each trace t_i consists of a justification j_i and an associated candidate verdict $v_i \in \{\texttt{True}, \texttt{False}, \texttt{Conflicting}\}$. We formulate the task as a reasoning trace selection problem. Rather than aggregating all candidate verdicts, the system ranks the candidate traces according to their estimated reliability. The highest-ranked trace is selected as the best trace:

$$t^* = \arg\max_{t_i \in T} s_i$$

where s_i denotes the reliability score assigned to trace t_i. The final prediction for the claim is the verdict associated with the selected trace:

$$\hat{v} = v^*$$

where v^* is the candidate verdict of t^*. Thus, the problem requires both ranking reasoning traces and producing the final claim-level verdict from the top-ranked trace.

4.2. TraceFT: A Fine-Tuned Verifier

We propose **TraceFT**, a fine-tuned LLM verifier for estimating the reliability of candidate reasoning traces. Rather than directly predicting the final claim label, TraceFT is trained to determine whether a given trace leads to the correct verdict.

For each candidate trace, we construct the input by combining the claim, evidence, candidate verdict, and justification:

$$x_i = [C; E; v_i; j_i]$$

where C is the claim, E is the evidence, v_i is the candidate verdict, and j_i is the reasoning justification. Before training, explicit label markers are removed from the justification text to reduce label leakage and encourage TraceFT to assess the quality of the reasoning itself.

To make fine-tuning more efficient, TraceFT applies LoRA instead of updating all parameters of the base LLM [18]. LoRA freezes the pretrained model weights and adds small trainable low-rank matrices to selected transformer layers. Given a pretrained weight matrix W, LoRA represents the update as a low-rank decomposition:

$$W' = W + \Delta W, \quad \Delta W = BA$$

where A and B are trainable low-rank matrices. During training, only the LoRA parameters and the classification head are updated, while the original LLM parameters remain frozen. This reduces memory and computational cost while still allowing the model to adapt to the reasoning trace verification task.

TraceFT is trained as a binary classifier. For each trace, the binary supervision label is automatically derived by comparing the candidate verdict v_i with the ground-truth verdict L:

$$y_i = \mathbb{1}[v_i = L]$$

where $y_i = 1$ indicates that the trace leads to the correct verdict, and $y_i = 0$ otherwise. TraceFT outputs a logit z_i, which is converted into a reliability score using the sigmoid function:

$$s_i = \sigma(z_i)$$

where s_i represents the estimated probability that trace t_i leads to the correct verdict.

During inference, TraceFT scores all candidate traces for the same claim independently. These scores are used to rank the traces, and the verdict associated with the highest-scoring trace is selected as the final prediction.

5. Experiment Setup

5.1. Dataset

For Task 2, the dataset consists of claims collected from multiple fact-checking domains. Each claim is paired with relevant evidence and a set of LLM-generated reasoning traces, where each trace is associated with a candidate verdict. To encourage diversity among candidate traces, the organizers generated multiple reasoning traces per claim using different temperature settings and applied de-duplication to remove redundant outputs. The full task dataset covers three languages: English, Spanish, and Arabic. In this work, we focus only on the English subset. Our experiments use the provided validation set for model development and the **English** test set for final evaluation.

5.2. Training and Inference

We select `meta-llama/Llama-3.2-3B`[1] as the base model for **TraceFT**. LLaMA-3.2-3B was selected over other available models in the LLaMA family due to its balance between capacity and cost. At 3B

[1] https://huggingface.co/meta-llama/Llama-3.2-3B

parameters, the model retains the general reasoning ability of the LLaMA-3 models but is still small enough to fine-tune under our computational limitations. Additionally, the model is openly available with instruction-tuned weights which supports the reproduction of our proposed system. TraceFT is fine-tuned using LoRA with rank 16, alpha 32, and dropout 0.05 applied to the query, key, value, and output projection layers. Inputs are tokenized to a maximum length of 512 tokens. We train TraceFT for 5 epochs with a batch size of 2 using AdamW with a learning rate of 1×10^{-4} and $\epsilon = 1 \times 10^{-8}$. The model is optimized with binary cross-entropy with logits loss, and a cosine learning rate schedule with a 5% warmup ratio is used. Gradients are clipped with a maximum norm of 1.0. The data is split into 80% training and 20% validation using stratified sampling. The checkpoint with the lowest validation loss is selected as the final TraceFT model. Table 1 presents the hyperparameter details.

Table 1
Training hyperparameters for TraceFT.

Hyperparameter	Value
Base model	`meta-llama/Llama-3.2-3B`
Maximum sequence length	512
Batch size	2
Number of epochs	5
Learning rate	1×10^{-4}
Optimizer	AdamW
AdamW epsilon	1×10^{-8}
Loss function	Binary cross-entropy with logits
Scheduler	Cosine schedule with warmup
Warmup ratio	5% of total training steps
Gradient clipping	Max norm 1.0
LoRA rank	16
LoRA alpha	32
LoRA dropout	0.05
LoRA target modules	`q_proj`, `k_proj`, `v_proj`, `o_proj`
Random seed	42

During inference, TraceFT scores all candidate reasoning traces for each claim independently. For each trace, TraceFT outputs a logit, which is converted into a probability score using the sigmoid function. This score represents the estimated reliability of the trace. The traces are then ranked in descending order by score, and the verdict associated with the highest-scoring trace is selected as the final prediction.

5.3. Evaluation

We evaluate our system using the official evaluation methodology of the CLEF 2026 CheckThat! Lab Task 2 [17]. The task is assessed as a combined ranking and classification problem. The quality of final claim verification is measured using macro-averaged F1 and class-wise F1 over the True, False, and Conflicting labels. Following the official setup, the final score is obtained by averaging the ranking across macro-F1 and Recall@5.

5.4. Baselines

To evaluate the effectiveness of **TraceFT**, we compare it with three baseline systems on the English test set. These baselines cover sparse retrieval with NLI classification, hybrid sparse–dense retrieval with NLI classification, and zero-shot LLM prediction.

BM25 Retrieval + NLI. This baseline first retrieves the most relevant evidence sentence using BM25 keyword matching [23]. The retrieved evidence and claim are then passed to a DeBERTa-v3 NLI model [24], which predicts one of three labels: entailment as `True`, contradiction as `False`, or neutral as

`Conflicting`. If no evidence is retrieved, the claim is labeled as `Conflicting`. This baseline tests whether direct evidence retrieval and NLI classification are sufficient for the task.

Hybrid Retrieval + NLI. This baseline extends BM25 retrieval by combining sparse lexical matching with dense semantic retrieval. Specifically, BM25 scores are combined with cosine similarity scores from a SentenceTransformer encoder using equal weighting:

$$\text{score}(e_i) = \alpha \cdot \text{BM25}(q, e_i) + (1 - \alpha) \cdot \cos(\mathbf{q}, \mathbf{e}_i),$$

where $\alpha = 0.5$. Both score vectors are normalized before combination. The highest-scoring evidence sentence is then classified using the same DeBERTa-v3 NLI model. This baseline evaluates whether combining lexical and semantic retrieval improves over BM25-only retrieval.

Zero-Shot LLM Classification. This baseline uses LLaMA-3.2-3B without fine-tuning. Each claim and its evidence are formatted into a prompt, and the model is instructed to output one of the three verdict labels: `True`, `False`, or `Conflicting`. Unlike TraceFT, this baseline directly predicts the claim-level verdict and does not rank reasoning traces. It is used to measure the out-of-the-box performance of the base LLM.

Zero-Shot LLM Classification Prompt

System message:
You are a fact-checking assistant. Given a claim and evidence, classify the claim as `True`, `False`, or `Conflicting`.

Label definitions:

- `True`: the evidence fully supports the claim.
- `False`: the evidence fully refutes the claim.
- `Conflicting`: the evidence is ambiguous or does not clearly support or refute the claim.

User message template:
Claim: `{claim}`
Evidence: `{evidence}`

Output instruction:
Answer with exactly one word: `True`, `False`, or `Conflicting`.

6. Results

Table 2 reports the evaluation results of TraceFT and the baseline systems on the English test set. Overall, **TraceFT** achieves the best performance among the reported systems, with an average score of **0.4261**, a macro-F1 score of **0.5988**, and a Recall@5 of **0.2534**, additionally ranking 2nd among all participating teams on the official CLEF 2026 leaderboard for the English numerical claims task, trailing the top-ranked system by only 0.0025 in average score (0.4286 vs 0.4261). Relative to the strongest baseline (Hybrid Retrieval + NLI), this corresponds to a 0.0231 gain in average score, though the macro-F1 margin is narrower at 0.0092. These results show that fine-tuning an LLM verifier for reasoning trace selection is effective for numerical claim verification.

Among the retrieval-based baselines, Hybrid Retrieval + NLI performs better than BM25 Retrieval + NLI, achieving a higher average score of 0.4030 compared with 0.3906 and a higher macro-F1 score of 0.5896 compared with 0.5668. This suggests that combining sparse lexical retrieval with dense semantic similarity provides a modest improvement over BM25-only retrieval. Hybrid Retrieval + NLI also improves performance on the `Conflicting` class, reaching an F1 score of 0.2342 compared with 0.1755 for BM25 Retrieval + NLI.

Table 2
Evaluation results of TraceFT and baselines on the English test set.

System	Avg. Score	Macro-F1	Recall@5	True F1	Conflicting F1	False F1
BM25 Retrieval + NLI	0.3906	0.5668	0.2144	0.6557	0.1755	0.8691
Hybrid Retrieval + NLI	0.4030	0.5896	0.2164	0.6589	0.2342	0.8757
Zero-Shot LLM Classification	–	0.0708	–	0.0069	0.1686	0.0369
TraceFT (ours)	**0.4261**	**0.5988**	**0.2534**	**0.7294**	**0.1788**	**0.8881**

The Zero-Shot LLM Classification baseline performs the worst, with a macro-F1 score of 0.0708. Its very low `True` F1 and `False` F1 scores indicate that zero-shot prompting without task-specific adaptation is not reliable for this task. The performance of the Zero-Shot baseline here when broken down into classes, where True F1 (0.0069) and False F1 (0.0369) are virtually zero, whereas Conflicting F1 (0.1686) is high in comparison provides insight to its performance. This class breakdown reveals how a foundational model struggles to commit to a definitive True or False verdict, thus defaulting to Conflicting for most claims likely due to the fact that verification requires precise numerical comparison of dates and quantities that the base Zero-Shot model cannot perform reliably from the prompt alone. This highlights the importance of fine-tuning and task-specific trace-level verification.

TraceFT achieves the strongest overall performance in comparison to the evaluated baseline systems and the best results on the `True` and `False` classes, with F1 scores of **0.7294** and **0.8881**, respectively. It also obtains the highest Recall@5 score, indicating that its verifier-based ranking is better at placing correct reasoning traces near the top. These results suggest that TraceFT learns useful trace-level reliability signals and can better select reasoning traces that lead to correct final verdicts.

However, the `Conflicting` class remains challenging. Although TraceFT improves overall performance, its `Conflicting` F1 score of 0.1788 is lower than that of Hybrid Retrieval + NLI. This suggests that ambiguous or incomplete evidence remains difficult for the verifier to identify. This weakness remains meaningful since in real-world fact-checking scenarios, correctly identifying conflicting evidence, where a claim may not be fully supported or refuted, is an important aspect of fact-checking. Future improvements should focus on handling ambiguous evidence, improving verifier calibration, and increasing training diversity for minority or difficult verdict classes.

7. Conclusion

In this paper, we presented **TraceFT**, a lightweight LLM fine-tuning approach for Task 2 of the CLEF 2026 CheckThat! Lab on numerical claim verification. We formulated the task as a reasoning trace selection problem, where each candidate trace is scored by a verifier and the verdict from the highest-scoring trace is used as the final prediction. By fine-tuning an LLM with LoRA, TraceFT learns trace-level reliability signals while keeping training computationally efficient. Experimental results show that TraceFT achieves the best performance among the evaluated systems, with an average score of 0.4261, a macro-F1 score of 0.5988, and a Recall@5 of 0.2534. On the official leaderboard, our system ranked 2nd among all participating teams on the English numerical claims task. Compared with retrieval-based and zero-shot LLM baselines, TraceFT demonstrates the effectiveness of fine-tuned verifier models for reasoning trace selection. In particular, it performs strongly on the `True` and `False` classes, showing that the verifier can identify useful reasoning patterns from candidate traces. Despite these results, the `Conflicting` class remains challenging, suggesting that ambiguous or incomplete evidence is still difficult for the verifier to handle.

For future work, we plan to improve TraceFT in three directions. First, we will address class imbalance through class-weighted loss, resampling, and calibration methods to improve performance on minority classes, specifically for the Conflicting class, since it was the hardest class for our verifier and also the most consequential class for practical fact-checking. Furthermore, we will explore giving

the verifier access to features that are more suited towards ambiguity, treating the Conflicting class as an important class, not a residual category. Second, we will explore stronger verifier training strategies, such as pairwise ranking losses and contrastive learning, to better distinguish high-quality and low-quality reasoning traces. Third, we will investigate larger or more specialized LLM backbones and longer input contexts to better preserve evidence and reasoning details. Currently, our method truncates all inputs to the maximum sequence length of 512 tokens, which could plausibly discard portions of longer reasoning traces or evidence sentences before they can be scored by the verifier. As such, claims which contain multi-part lengthy evidence are the most affected. In the future, we plan to investigate increasing the maximum sequence length to better preserve evidence and reasoning details. Finally, we will extend TraceFT to the Spanish and Arabic subsets to determine whether or not verifier-based trace selection transfers across languages, since our current findings are only limited to English. These directions may further improve the robustness of fine-tuned LLM verifiers for numerical and temporal claim verification.

Declaration on Generative AI

During the preparation of this work, the author(s) used AI tools for grammar and language refinement. These tools were employed to refine sentence structure, correct typographical errors, and improve overall language quality. No generative content was used for analysis, figures, or experimental sections. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] R. Aly, Z. Guo, M. Schlichtkrull, J. Thorne, A. Vlachos, C. Christodoulopoulos, O. Cocarascu, A. Mittal, FEVEROUS: Fact extraction and VERification over unstructured and structured information, in: Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1 (NeurIPS Datasets and Benchmarks 2021), 2021. URL: https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/68d30a9594728bc39aa24be94b319d21-Abstract-round1.html.

[2] L. Allein, M. Saelens, R. Cartuyvels, M. F. Moens, Implicit temporal reasoning for evidence-based fact-checking, in: Findings of the Association for Computational Linguistics: EACL 2023, 2023, pp. 176–189.

[3] A. M. Barik, W. Hsu, M. L. Lee, Time matters: An end-to-end solution for temporal claim verification, in: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track, Association for Computational Linguistics, 2024, pp. 657–664. URL: https://aclanthology.org/2024.emnlp-industry.48.

[4] M. Strong, A. Vlachos, Tsver: A benchmark for fact verification against time-series evidence, in: Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, 2025, pp. 29894–29914.

[5] V. Venktesh, D. Prabhu, A. Anand, A benchmark for open-domain numerical fact-checking enhanced by claim decomposition, arXiv preprint arXiv:2510.22055 (2025).

[6] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al., Chain-of-thought prompting elicits reasoning in large language models, Advances in neural information processing systems 35 (2022) 24824–24837.

[7] 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 (2023). URL: https://openreview.net/forum?id=1PL1NIMMrw.

[8] T. Lanham, A. Chen, A. Radhakrishnan, B. Steiner, C. Denison, D. Hernandez, D. Li, E. Durmus, E. Hubinger, J. Kernion, et al., Measuring faithfulness in chain-of-thought reasoning, arXiv preprint arXiv:2307.13702 (2023).

[9] K. Kim, S. Lee, K.-H. Huang, H. P. Chan, M. Li, H. Ji, Can llms produce faithful explanations for fact-checking? towards faithful explainable fact-checking via multi-agent debate, arXiv preprint arXiv:2402.07401 (2024).
[10] X. Wang, et al., Self-consistency improves chain of thought reasoning in language models, arXiv preprint arXiv:2301.03127 (2023). URL: https://arxiv.org/abs/2301.03127.
[11] 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, arXiv preprint arXiv:2110.14168 (2021).
[12] 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, in: International Conference on Learning Representations, volume 2024, 2024, pp. 39578–39601.
[13] A. Setlur, C. Nagpal, A. Fisch, X. Geng, J. Eisenstein, R. Agarwal, A. Agarwal, J. Berant, A. Kumar, Rewarding progress: Scaling automated process verifiers for llm reasoning, in: International Conference on Learning Representations, volume 2025, 2025, pp. 60808–60838.
[14] J. M. Struß, S. Schellhammer, S. Dietze, V. V., V. Setty, T. Chakraborty, P. Nakov, A. Anand, P. Chungkham, S. Hafid, D. Sahnan, K. Todorov, The clef-2026 checkthat! lab: Advancing multilingual fact-checking, in: R. Campos, A. Jatowt, Y. Lan, M. Aliannejadi, C. Bauer, S. MacAvaney, A. Anand, Z. Ren, S. Verberne, N. Bai, M. Mansoury (Eds.), Advances in Information Retrieval, Springer Nature Switzerland, Cham, 2026, pp. 325–335.
[15] J. M. Struß, S. Schellhammer, S. Dietze, V. V., V. Setty, T. Chakraborty, P. Nakov, A. Anand, P. Chungkham, S. Hafid, D. Sahnan, K. Todorov, Overview of the CLEF-2026 CheckThat! Lab: Advancing multilingual fact-checking, in: M. Hagen, M. Potthast, B. Stein, P. Schaer, E. Zangerle, S. MacAvaney, J. M. Struß, E. S. Salido, A. Barrón-Cedeño, Alba García Seco de Herrera (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Seventeenth International Conference of the CLEF Association (CLEF 2026), Springer, 2026.
[16] E. S. Salido, A. Barrón-Cedeño, Alba García Seco de Herrera, S. MacAvaney, J. M. Struß (Eds.), CLEF 2026 Working Notes, CLEF 2026, Jena, Germany, 2026.
[17] V. V, V. Setty, P. Chungkham, A. Anand, F. Alam, Overview of the CLEF-2026 CheckThat! lab task 2 on fact-checking numerical claims, in: [16], 2026.
[18] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, Lora: Low-rank adaptation of large language models, in: International Conference on Learning Representations, 2022. URL: https://openreview.net/forum?id=nZeVKeeFYf9.
[19] Z. Guo, M. Schlichtkrull, A. Vlachos, A survey on automated fact-checking, Transactions of the association for computational linguistics 10 (2022) 178–206.
[20] 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.
[21] D. Wadden, S. Lin, K. Lo, L. L. Wang, M. van Zuylen, A. Cohan, H. Hajishirzi, Fact or fiction: Verifying scientific claims, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 7534–7550.
[22] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, et al., The llama 3 herd of models, arXiv preprint arXiv:2407.21783 (2024).
[23] S. E. Robertson, S. Walker, Some simple effective approximations to the 2–poisson model for probabilistic weighted retrieval, in: SIGIR'94: Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, organised by Dublin City University, Springer, 1994, pp. 232–241.
[24] P. He, J. Gao, W. Chen, DeBERTaV3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing, in: The Eleventh International Conference on Learning Representations, 2023. URL: https://openreview.net/forum?id=sE7-XhLxHA.