AlexU CSED Digi at CheckThat! 2026: Generative Verifier and Fused Reranker-Verifier for Fact-Checking Numerical Claims

Notebook for the CheckThat! Lab at CLEF 2026

Mahmoud M. Abdelaziz[1], Mohamed M. Masseoud[1], Marwan Torki[1] and Nagwa El-Makky[1]

[1]Department of Computer and Systems Engineering, Alexandria University, Egypt

Abstract

The growing spread of misinformation, particularly claims including numerical information, has increased the demand for trustworthy automated fact-checking systems capable of precise numerical reasoning. This paper presents our approach to Task 2 of the CheckThat! 2026 lab on multilingual numerical fact-checking. We propose two approaches for training a verifier to score reasoning traces generated for test-time scaling: (i) a decoder-only LLM generative verifier fine-tuned for next-token prediction objective to score reasoning traces via Yes/No token prediction, and (ii) an ensemble that fuses a reranker trained with listwise loss with a claim verifier trained with weighted cross-entropy loss via Reciprocal Rank Fusion. We evaluate our approaches on three languages: Arabic, English, and Spanish. On the test set, our generative verifier achieves a test score of 0.5876, ranking 5th on the Arabic leaderboard, and 0.4080 on the English leaderboard, our fused reranker-verifier achieves score 0.4189 on Spanish test set, ranking 7th. We conduct ablation experiments on development sets to investigate how model scale, context length, LoRA adapter parameters and fusion strategy affect scores.

Keywords

Numerical Fact-Checking, Reasoning Trace Reranking, Test-Time Scaling, Generative Verifiers, Listwise Reranking, Multilingual NLP

1. Introduction

Automated fact verification has made substantial progress in recent years, supported by large-scale benchmarks designed to detect and mitigate misinformation[1, 2, 3, 4]. However, the task remains inherently knowledge-intensive, requiring reasoning over heterogeneous evidence sources to reach a reliable verdict. This becomes especially difficult for claims involving numerical or temporal information, where correct interpretation depends on precise quantitative reasoning rather than surface-level semantic matching.

Motivated by recent advances in test-time reasoning for numerical fact checking[5], the CLEF 2026 CheckThat! lab [6] introduced task 2 [7], a test-time scaling setting for fact verification. Given a claim, supporting evidence, and multiple reasoning traces generated by a large language model (LLM) each reaching a verdict, the goal is to rank these traces by their usefulness in producing the correct verdict and use the highest-ranked traces to derive a final prediction. The challenge is to train a verifier that assigns scores to each reasoning trace and ranks them based on that scoring.

To address this setting, we explore two verifier-training approaches. The first is a fine-tuned decoder-only LLM. Inspired by the recent work on generative verifiers [8], we reformulate the ranking task as auto-regressive token generation, leveraging pretrained language understanding without architectural modification. The second is an ensemble approach. We fuse the rankings from an encoder-based reranker, and encoder-based claim verifier, the reranker is trained with listwise loss to score the reasoning traces and the claim verifier is trained with a weighted cross-entropy loss to classify the claims, and rankings from both are fused using Reciprocal Rank Fusion to produce the final traces'

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

es-MahmoudM.Abd-Elaziz2024@alexu.edu.eg (M. M. Abdelaziz); es-MohamedM.Rashad2024@alexu.edu.eg (M. M. Masseoud); mtorki@alexu.edu.eg (M. Torki); nagwamakky@alexu.edu.eg (N. El-Makky)

Table 1
Train set label distribution per language.

Language	False	True	Conflicting	Total
Arabic	1,444	1,164	—	2,608
English	3,717	1,175	1,508	6,400
Spanish	1,888	152	206	2,246

ranking. The final predictions are derived from the top-ranked reasoning trace for both approaches using Best-of-N (BoN).
Our main contributions are as follows:

1. We investigate generative verifiers for multilingual numerical fact verification by reformulating reasoning-trace evaluation as a next-token prediction problem. Our experiments show that generative verifiers fine-tuned with a cross-entropy objective can achieve competitive performance, with Qwen3-4B [9] and Llama-3.2-3B [10] producing our strongest official leaderboard results on the Arabic and English test sets, respectively.
2. We propose an ensemble approach that combines an encoder-based reranker trained with a listwise ranking objective and an encoder-based claim verifier trained with weighted cross-entropy loss. We fuse their rankings using Reciprocal Rank Fusion (RRF), obtaining higher scores than baseline on Arabic and Spanish and better Recall in English.
3. We train and evaluate both approaches independently for English, Spanish, and Arabic, and report comprehensive development-set ablations together with official leaderboard results.
4. We analyze the effects of model scale, LoRA rank, context length, training objective, and fusion strategy on multilingual reasoning-trace verification, highlighting the different generalization behavior when models are trained with different losses.

The remainder of this paper is organized as follows. Section 2 describes the task and dataset. Section 3 reviews related work. Section 4 presents our proposed methods. Section 5 details experimental setup. Section 6 discusses results, ablations, and analysis. Section 7 concludes the paper.

2. Task and Dataset

2.1. Task Definition

The CLEF 2026 CheckThat! Lab task 2 focuses on test-time scaling for fact-verification. Given a claim c, a set of retrieved evidence documents $D = \{d_1, \ldots, d_n\}$, and 20 reasoning traces $T = \{(j_i, v_i)\}_{i=1}^{20}$ generated by GPT-4o-mini. The task is to (1) train a verification model, to select the best reasoning traces that lead to the correct verdict. (2) predict the final claim verdict label $v \in \{\text{True}, \text{False}, \text{Conflicting}\}$. Each reasoning trace includes a predicted verdict v_i and a justification j_i.

Formally, a verifier model f_θ produces a relevance score $s(t \mid c, D)$ for each trace $t \in T$, from which a ranked list and final verdict prediction are derived. generative verifiers (GV) and reranker consume a concatenated input: `[claim]` $\oplus$ `[evidence]` $\oplus$ `[verdict]` $\oplus$ `[justification]`, while the claim verifier in the fused reranker-verifier ensemble approach is not fed with the verdict.

2.2. Dataset and Metrics

The datasets consist of 8,000 English claims released in [11], 2,808 Spanish claims, and 3,260 Arabic claims from CLEF 2025 [12], each with 20 reasoning traces and retrieved evidence. Each reasoning trace has a final verdict, which takes one of three values True, False, and Conflicting. Exploratory data analysis revealed that the Arabic dataset lacks the Conflicting class, and that more than half of the data lack opposite signal traces, Table 2 shows the percentage of data samples where all reasoning

traces agreed on the same verdict, for the Arabic dataset 57.89% of the data lacks a reasoning trace with opposing verdict, hindering pairwise-loss-based training.

Table 2
Training set statistics showing the percentage of instances where all 20 generated traces were correct, all were wrong, or mixed.

Language	All Verdicts Correct (%)	All Verdicts Wrong (%)	Mixed Verdicts (%)
Arabic	48.27	9.62	42.11
English	23.83	19.67	56.50
Spanish	45.68	13.40	40.92

Trace ranking quality is measured using **Recall@5**. After ranking traces for each claim with the verifier, Recall@5 is defined as

$$\text{Recall@5} = \frac{\#\{\text{claims with at least one relevant trace in the top-5}\}}{\#\{\text{claims}\}}.$$

where relevant trace is a trace with correct final verdict. Claim verification performance is evaluated using **Macro-F1**, which is the unweighted average of the F1-score computed independently for each claim class:

$$\text{Macro-F1} = \frac{1}{C}\sum_{i=1}^{C} F1_i,$$

where C is the number of classes (True, False, and Conflicting). The official leaderboard score averages these two metrics:

$$\text{Score} = \frac{\text{Macro-F1} + \text{Recall@5}}{2}.$$

We observed that, for the Arabic dataset, the leaderboard computes Macro-F1 as

$$\text{Macro-F1} = \frac{F1_{\text{True}} + F1_{\text{False}}}{2},$$

excluding the Conflicting class. We therefore use the same formulation in our experiments for consistency with the official evaluation.

3. Related Work

Claim verification has been a core task across multiple CheckThat! editions. Task 3 of the 2025 edition focused on numerical claim verification. The 2026 edition shifts focus to test-time reasoning trace evaluation, introducing a two-stage paradigm where traces are generated by an LLM and evaluated by a separately trained verifier. Fact-checking complex numerical claims often requires multi-step reasoning, an area where standard large language models struggle due to reasoning drift. To address this, Chungkham et al.[5] explored scaling test-time compute (TTS) to elicit multiple reasoning paths. They introduced VerifierFC, a model trained to score and select optimal reasoning traces via BoN selection. Their adaptive TTS approach mitigates reasoning drift and yielded substantial performance improvements over single-shot methods, specifically 18.8% on the QuanTemp dataset[11] and 21.27% on the ClaimDecomp dataset[13].

Recent work [8] shows that training verifiers using next-token prediction yields superior performance compared to discriminative methods. The resulting generative verifiers align naturally with instruction tuning and scale with model size and test-time compute.

Listwise ranking objectives have proven effective for document retrieval and re-ranking. ListMLE [14] and ListNet [15] optimize ranking metrics directly, offering an alternative to pointwise training for

trace scoring. Reciprocal Rank Fusion [16] provides a principled method for combining ranked lists from heterogeneous models without requiring score calibration. Our second approach integrates these components into a unified ensemble.

4. Methodology

4.1. Approach 1: Generative LLM Verifier

Rather than appending a task-specific classification head, we fine-tune a decoder-only LLM to evaluate the reasoning trace via next-token prediction. We structure the input sequence as $x = \texttt{[prompt]} \oplus \texttt{[claim]} \oplus \texttt{[evidences]} \oplus \texttt{[verdict]} \oplus \texttt{[justification]}$, where the system prompt instructs the model to act as a fact-checking auditor evaluating the trace: *"You are an expert fact-checking auditor. Your job is to evaluate whether a previous fact-checker's verdict and justification are correct given the claim and retrieved documents. You only respond with Yes or No: Yes if the claim checker's verdict and justification are correct, No if they are incorrect."*

Based on this input, we extract the model's unnormalized logits specifically for the "`Yes`" and "`No`" tokens. The trace verification score is defined strictly as the raw logit of the Yes token:

$$\text{score}(x) = z_{\text{Yes}}$$

where z_{Yes} is the unnormalized logit of the Yes token. To determine the final claim verdict, we use a BoN selection strategy, the model is fine-tuned using cross-entropy loss, where the target token is "`Yes`" for reasoning traces that have a correct verdict and "`No`" otherwise.

This reframes ranking as a generation task, leveraging pre-trained language understanding without architectural modification. All models are fine-tuned with Low-Rank Adaptation (LoRA) [17] applied to query, key, and value projections with $\alpha = 2 \times r$.

Training Details The models were trained only on the provided datasets, without augmentation, separately on Arabic, English, and Spanish. A shared setup between experiments is (learning rate 10^{-4}, batch size of 32–128, trained for 2–3 epochs, and optimizer is AdamW with weight decay 0.01).

4.2. Approach 2: Discriminative Reranker–Verifier Ensemble

We decompose the problem into two subtasks: trace scoring (reranker) and claim verification without the trace (verifier). The **reranker** is based on mDeBERTa-v3-base. It encodes each claim, evidence set, and candidate trace, then assigns a scalar score to it. Traces with verdict matching the gold label are treated as positives, while the remaining traces are negatives. The reranker is trained with a compound loss of Multi-Positive ListNet and pairwise ranking loss, with both losses having equal weights 0.5.

The **claim verifier** is based on XLM-RoBERTa-large. It receives only the claim and evidence, without the generated traces, and predicts one of the verdicts: *False*, *True*, or *Conflicting*, the claim verifier is trained with weighted cross-entropy loss, where the class weights are inversely proportional to the class frequencies. At inference time, rankings from both reranker and verifier are fused via RRF.

4.2.1. Fusion Strategies

We evaluated z-score fusion, Reciprocal Rank Fusion (RRF), and logistic regression fusion. Z-score fusion combines normalized reranker scores with verifier log-probabilities:

$$s_i = z(r_i) + \alpha \log P_{\text{verifier}}(v_i \mid c, D) + b_{v_i}. \tag{1}$$

RRF combines the ranks produced by the reranker and verifier:

$$s_i = \frac{w_r}{k + \text{rank}_r(i)} + \frac{w_v}{k + \text{rank}_v(i)} + b_{v_i}. \tag{2}$$

where $k = 5$, $w_r = 1.0$, $w_v = 1.5$, and b_{v_i} is a class-bias term.

Logistic regression fusion model is trained on reranker scores and verifier probabilities.

Training Details. LoRA fine-tuning is used for both models with rank ($r = 32$), the maximum context length for reranker is 1024 and 512 for claim verifier. For Spanish and Arabic, we initialize from the English-trained checkpoints and continue fine-tuning on the target-language data. Spanish keeps the same three-class setup as English. For Arabic, the gold labels contain only *False* and *True*, but traces may still contain *Conflicting*; therefore, we keep the three-logit verifier head for compatibility and apply a negative Conflicting bias during fusion. All models were trained only on the provided task datasets, without augmentation. We used AdamW with weight decay 0.01, learning rate 10^{-4}, batch size 64–128, bfloat16 precision, and 2–3 epochs. For Spanish and Arabic transfer, we continued from the English checkpoints with a lower learning rate and trained for 4 epochs.

For both approaches, LoRA adapters target query, key, and value projections with $\alpha = 2 \times r$.

5. Experimental Setup

Training was done on Google Colab and vast.ai machines, NVIDIA H100 (80 GB) and RTX 5090 (32 GB) GPUs.

6. Results and Analysis

In this section, we present the results of our two approaches, followed by ablation experiments for both.

6.1. Results on Development Set

First Approach: Generative Verifier: We first fine-tuned the baseline provided by the competition (Llama-3.2-1B with pointwise Binary Cross Entropy loss and context window 256) to serve as our baseline on the Arabic, English, and Spanish datasets. Then, we experimented with the same model and context window but with a generative verifier setup. Observing an increase in Macro-F1 and Recall@5 from the generative verifier approach on the Arabic dataset, we decided to use a larger model: Qwen3-4B. The model was fine-tuned in "non-thinking" mode by providing the `/no_think` token in the system prompt. Although Qwen3-4B is substantially larger than Llama-3.2-1B, it did not consistently improve development set performance across the three languages, which motivated us to explore other approaches for training verifiers. We therefore trained an encoder-based reranker using a listwise ranking loss.

Second Approach: Reranker-Verifier Ensemble: We first fine-tuned mDeBERTa-v3-base [18] for trace scoring with Multi-positive ListNet + Pairwise ranking loss, achieving results comparable to baseline and better than the generative verifier on development set, we experimented with training an XLM-RoBERTa-large [19] based claim verifier and achieved similar results, so rank fusion was applied to both rankings and it yielded better scores than either individual model, this approach achieved the better scores than the baseline on development set in the Arabic and Spanish and close to baseline on English, the results for both approaches and the baseline are shown in Table 3.

6.2. Official Leaderboard Results

Table 4 presents official test set results. The final submitted systems were Qwen3-4B Generative verifier with LoRA rank 64 for Arabic, with a leaderboard score 0.5876 ranking 5th, only 0.0167 behind the first place. For Spanish, we submitted the results of Reranker–Verifier Ensemble ranking 7th on leaderboard. For English, fine-tuning Llama-3.2-3B with the same generative setup as Arabic yielded a slightly better score than Qwen3-4B (0.4080 vs 0.4060) so this model was submitted for English.

Table 3
Comparison of our proposed approaches against the baseline on development sets for Arabic, Spanish, and English.

Dev set	System	Recall@5	Macro-F1	Score
Arabic	Baseline	0.2705	0.7921	0.5313
	Qwen3-4B GV	0.2166	0.7534	0.4850
	Ensemble RRF	**0.2712**	**0.7950**	**0.5331**
Spanish	Baseline	0.2221	0.3724	0.3204
	Qwen3-4B GV	0.2030	0.3815	0.2922
	Ensemble RRF	**0.2579**	**0.4767**	**0.3673**
English	Baseline	0.3031	**0.5263**	**0.4147**
	Qwen3-4B GV	0.2664	0.4858	0.3761
	Ensemble RRF	**0.3076**	0.5128	0.4102

Table 4
Official leaderboard results.

Language	Rank	System Approach	Macro-F1	Recall@5	Score
Arabic	5th	Qwen3-4B GV	0.8430	0.3322	0.5876
Spanish	7th	Ensemble RRF	0.5920	0.2457	0.4189
English	16th	Llama-3.2-3B GV	0.5971	0.2188	0.4080

6.3. Ablation experiments

Ablations on Generative Verifier We conducted an ablation on the generative verifier approach, comparing training with a linear head and pointwise binary cross-entropy loss (BCE) against generative cross-entropy loss (CE) over Yes/No tokens, varying LoRA rank ($r \in \{8, 16, 128\}$), context window (256–1024 tokens), and model. Table 5 reports results.

Table 5
Approach 1 ablation on the development sets.

Model	Loss	r	Context	Arabic		Spanish		English	
				Rec@5	MF1	Rec@5	MF1	Rec@5	MF1
Context Length									
Llama-3.2-1B	BCE	8	256	0.2705	0.7921	0.2221	0.3724	0.3031	0.5263
Llama-3.2-1B	BCE	8	1024	0.2031	0.7414	0.2052	0.3819	0.2412	0.4618
Llama-3.2-1B	CE	8	256	0.2747	0.8012	0.2662	0.4001	0.3017	0.5056
Llama-3.2-1B	CE	8	1024	0.2166	0.7534	0.2685	0.397	0.3012	0.5096
LoRA rank									
Llama-3.2-1B	BCE	128	1024	**0.2769**	**0.8014**	0.2042	0.3663	0.2436	0.4731
Llama-3.2-1B	CE	128	1024	0.2209	0.7598	**0.2759**	0.4464	**0.3083**	**0.5274**
Model Scaling									
Qwen3-4B	CE	16	1024	0.2166	0.7546	0.2109	0.3936	0.3044	0.5065
Qwen3-4B	CE	64	1024	0.2166	0.7607	0.2723	**0.4575**	0.2645	0.4875

The following findings emerge: In our experiments, increasing the backbone size from Llama-3.2-1B to Qwen3-4B did not consistently improve performance across the three development sets. Although Qwen3-4B achieves the highest Macro-F1 in Spanish, it underperforms Llama-3.2-1B on Arabic and English.

Longer context windows provide limited benefits. Using a maximum context length of 1024 does not consistently outperform a 256-token context window. For Llama-3.2-1B with LoRA rank 8, the shorter

context generally achieves better results under both BCE and CE losses. This suggests that additional context may introduce noise that outweighs the benefit of preserving the complete reasoning trace. Nevertheless, the concatenation of the system prompt, claim, evidence, reasoning trace, and verdict can exceed 512 tokens, meaning that the 256-token setting may truncate part of the reasoning process.
The effect of LoRA rank depends on the training objective. Increasing the LoRA rank from 8 to 128 has different effects under the BCE and CE objectives. Models trained with the generative CE objective generally benefit more from larger LoRA ranks across languages, with the exception of Qwen3-4B on the English dataset. In contrast, models trained with a linear classification head and BCE loss exhibit mixed results, indicating that higher adapter capacity is utilized more effectively under the generative training objective.

Overall, these results suggest that increasing model capacity alone or increasing context window, is insufficient for consistently improving generative verifier performance.

We also analyzed fusion strategies for the second approach (Table 6). Despite z-score fusion leading on the English and Arabic development sets. RRF achieved the strongest performance on the official test set. We attribute this difference to the fact that z-score fusion relies on calibration parameters (α, `true_bias`, `conf_bias`) that are optimized on the development set distribution and may overfit to it. In contrast, RRF fusion operates on ordinal ranks rather than raw score magnitudes, making it more robust to distributional shift between development and test data.

Table 6
Approach 2 fusion ablation on Dev sets.

	English		Arabic		Spanish	
System	Recall@5	Macro-F1	Recall@5	Macro-F1	Recall@5	Macro-F1
Reranker only	0.2236	0.4539	0.2394	0.4989	0.2572	0.3698
Verifier only	<u>0.2976</u>	0.5035	0.2757	<u>0.5337</u>	0.2626	**0.4224**
Logistic regression	**0.3026**	0.5000	**0.2766**	0.5336	**0.2764**	0.3801
Z-score fusion	0.2736	**0.5086**	<u>0.2760</u>	**0.5347**	<u>0.2720</u>	0.4061
RRF fusion	0.2866	<u>0.5064</u>	0.2712	0.5305	0.2705	<u>0.4204</u>

7. Conclusion and Future Work

We presented two approaches for fact-checking numerical claims through reasoning trace reranking at CheckThat! 2026 Task 2. The first approach showed that the generative CE objective makes more effective use of higher LoRA ranks than BCE training, with Qwen3-4B generative verifier as our submitted system for Arabic and Llama-3.2-3B for English. The second approach showed that a reranker-verifier ensemble fused via RRF captures complementary ranking signals, outperforming either component alone, and outperforming generative verifier on the development set serving as our submitted system for Spanish. On the official leaderboard, we placed 5th on Arabic, 7th on Spanish, and 16th on English.

Neither paradigm dominates consistently across all three languages and all dataset splits. Future work includes exploring transfer learning for generative verifiers by performing aggregate fine-tuning on the English, Arabic, and Spanish datasets, exploring data augmentation, such as generating opposite reasoning traces for samples where all reasoning traces have the same final verdict, and exploring pairwise training objectives.

Declaration on Generative AI

During the preparation of this work, the authors used Claude (Anthropic) and Gemini (Google) in order to do: Identify and correct grammar errors and typos, and to rephrase sentences to improve quality. After

using these tools, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] Z. Guo, M. Schlichtkrull, A. Vlachos, A survey on automated fact-checking, Transactions of the association for computational linguistics 10 (2022) 178–206.

[2] V. Setty, Factcheck editor: Multilingual text editor with end-to-end fact-checking, in: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2024, pp. 2744–2748.

[3] I. Augenstein, C. Lioma, D. Wang, L. C. Lima, C. Hansen, C. Hansen, J. G. Simonsen, Multifc: A real-world multi-domain dataset for evidence-based fact checking of claims, in: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP), 2019, pp. 4685–4697.

[4] M. Schlichtkrull, Z. Guo, A. Vlachos, Averitec: A dataset for real-world claim verification with evidence from the web, Advances in Neural Information Processing Systems 36 (2023) 65128–65167.

[5] P. Chungkham, V. Viswanathan, V. Setty, A. Anand, Think right, not more: Test-time scaling for numerical claim verification, in: Empirical Methods in Natural Language Processing (EMNLP) 2025, Association for Computational Linguistics, 2025, pp. 24345–24363.

[6] 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.

[7] 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: 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.

[8] L. Zhang, A. Hosseini, H. Bansal, S. M. Kazemi, A. Kumar, R. Agarwal, Generative verifiers: Reward modeling as next-token prediction, in: International Conference on Learning Representations (ICLR), 2025.

[9] A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv, et al., Qwen3 technical report, arXiv preprint arXiv:2505.09388 (2025).

[10] 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).

[11] V. V, A. Anand, A. Anand, V. Setty, Quantemp: A real-world open-domain benchmark for fact-checking numerical claims, in: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '24, Association for Computing Machinery, New York, NY, USA, 2024, p. 650–660. URL: https://doi.org/10.1145/3626772.3657874. doi:10.1145/3626772.3657874.

[12] F. Alam, J. M. Struß, T. Chakraborty, S. Dietze, S. Hafid, K. Korre, A. Muti, P. Nakov, F. Ruggeri, S. Schellhammer, et al., The clef-2025 checkthat! lab: Subjectivity, fact-checking, claim normalization, and retrieval, in: European Conference on Information Retrieval, Springer, 2025, pp. 467–478.

[13] J. Chen, A. Sriram, E. Choi, G. Durrett, Generating literal and implied subquestions to fact-check complex claims, in: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, 2022, pp. 3495–3516.

[14] F. Xia, T.-Y. Liu, J. Wang, W. Zhang, H. Li, Listwise approach to learning to rank: theory and algorithm, in: Proceedings of the 25th international conference on Machine learning, 2008, pp. 1192–1199.

[15] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, H. Li, Learning to rank: from pairwise approach to listwise approach, in: Proceedings of the 24th international conference on Machine learning, 2007, pp. 129–136.

[16] G. V. Cormack, C. L. Clarke, S. Buettcher, Reciprocal rank fusion outperforms condorcet and individual rank learning methods, in: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, 2009, pp. 758–759.

[17] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, et al., Lora: Low-rank adaptation of large language models., Iclr 1 (2022) 3.

[18] P. He, J. Gao, W. Chen, Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing, arXiv preprint arXiv:2111.09543 (2021).

[19] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, in: Proceedings of the 58th annual meeting of the association for computational linguistics, 2020, pp. 8440–8451.