Leifus at CheckThat! 2026: A Lightweight Verifier for Multilingual Reasoning-Trace Ranking

Hannan Shen[1], Hui Chen[1]

[1]Foshan University, Foshan, China

Abstract

This paper describes a verifier-based system for Task 2 of the CheckThat! Lab at CLEF 2026, which asks participants to rank generated reasoning traces for numerical claim verification and to derive the final veracity label from the best-ranked trace. We model the task as reward-style binary verification: each candidate reasoning trace is paired with the claim and its proposed verdict, and a classifier estimates whether the trace leads to the gold verdict. We compare DeBERTa-v3, XLM-RoBERTa-large, LLaMA-3.2-1B, and LLaMA-3.2-3B with LoRA or QLoRA adaptation, and we evaluate binary cross-entropy, full fine-tuning, pairwise ranking, pooling variants, and top-k verdict aggregation. In local evaluation with the official scorer and released labels, the best English run achieved 0.2547 Recall@5 and 0.8162 accuracy using LLaMA-3.2-1B LoRA, while the best Arabic and Spanish runs used LLaMA-3.2-3B QLoRA with 0.2876 and 0.2727 Recall@5, respectively. Additional analysis shows that majority voting over the top-ranked traces can improve macro-F1 in English and Spanish, mainly by recovering some Conflicting claims, but often reduces accuracy. All reported scores are offline post-evaluation results computed with the released labels and scorer, and are intended for within-paper comparison rather than official leaderboard comparison. Our findings suggest that compact decoder verifiers transfer well, while the Conflicting class remains the main bottleneck for robust macro-F1.

Keywords

CLEF CheckThat!, Numerical claims, Reasoning-trace ranking, Verifier model, Multilingual fact-checking, LoRA, QLoRA

1. Introduction

Numerical and temporal claims are common in political, financial, health, and social-media discourse. They are difficult to verify automatically because correctness can depend on quantities, dates, units, comparison operators, and the context in which a number was reported. Changing a time period, omitting a denominator, or comparing incompatible populations can turn an apparently plausible statement into a false or misleading one. The CheckThat! shared tasks have therefore devoted increasing attention to numerical claim verification, including the QuanTemp benchmark and the CLEF 2025 numerical-claim task [8, 12]. The 2026 CheckThat! Lab continues this direction with a test-time-scaling setup for multilingual fact verification [3, 1].

Task 2 differs from a conventional three-way veracity classification task. For each claim, the input contains multiple candidate reasoning traces and verdicts generated by a large language model. Participants must rank the traces according to their usefulness for arriving at the correct verdict, and the final claim label is derived from the highest-ranked trace. The official evaluation combines ranking quality, measured by Recall@5, with classification quality, measured by macro-F1 [1, 2]. This formulation shifts the modeling problem from generating new rationales to learning a verifier that can score candidate rationales.

Our system is deliberately lightweight. Instead of calling proprietary LLMs at inference time, we fine-tune open-source encoders and decoders with parameter-efficient adaptation. We cast the problem as binary trace verification: a trace is positive when its proposed verdict agrees with the gold claim label and negative otherwise. At inference time, the verifier score is used directly as the ranking score. By unified, we mean that the same trace-centered input representation, binary target construction, and

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

shnlfds@gmail.com (H. Shen); chenhui@fosu.edu.cn (H. Chen)

scoring rule are used across English, Spanish, and Arabic, although the underlying backbone models may differ.

We organize the experiments around three research questions: RQ1 asks which lightweight verifier architecture is most effective for multilingual trace ranking; RQ2 asks how training objectives and pooling choices affect Recall@5 and final verdict prediction; and RQ3 asks when top-k verdict aggregation improves macro-F1 and what errors remain for Conflicting claims.

The main contributions of this work are as follows. First, we provide a unified verifier formulation for trace scoring in a multilingual numerical fact-checking task. Second, we compare encoder, decoder, LoRA, QLoRA, full fine-tuning, and pairwise-ranking variants under the same scoring procedure. Third, we analyze English, Arabic, and Spanish results with preprocessing statistics, class-wise F1, top-k verdict aggregation, and a focused error analysis of the Conflicting class.

2. Related Work

Automated fact-checking pipelines usually combine retrieval, evidence selection, and veracity prediction. Early claim-verification benchmarks such as FEVER established the evidence-based classification setting over textual sources, while TabFact extended verification to semi-structured tabular evidence and highlighted the need for symbolic and numerical reasoning [9, 10]. Numerical claims add further challenges because small changes in quantities or time spans can invert the verdict [8]. The QuanTemp benchmark emphasized real-world open-domain numerical fact-checking and helped motivate later CheckThat! evaluations focused on quantities and temporal expressions [8]. In contrast to retrieval-heavy pipelines, Task 2 provides candidate reasoning traces and asks participants to identify the most reliable traces; this changes the emphasis from evidence acquisition to verifier learning.

CheckThat! 2025 Task 3 showed that transformer PLMs and LLM-based systems can be effective for multilingual numerical fact verification [12]. The strongest 2025 systems often relied on multi-stage retrieval and reasoning pipelines. For example, LIS used open-source LLMs for question generation, retrieval, and LoRA-based veracity prediction [13]; NGU_Research combined dense retrieval, BM25, and LLM judging [14]; and ClaimIQ compared prompted LLMs with supervised LoRA fine-tuning [15]. Recent complex-claim fact-checking work also uses explicit reasoning programs to decompose claims into sub-tasks, showing the importance of inspectable intermediate reasoning [11]. Unlike these systems, our setting assumes candidate traces are already available and focuses on ranking them.

Our work follows the supervised adaptation line but adapts it to the 2026 trace-ranking setup. Methodologically, our experiments use LoRA [17] and QLoRA [18] to adapt DeBERTa-v3 [19], XLM-RoBERTa [20], and LLaMA-3.2 [21]. All models are implemented with the HuggingFace Transformers ecosystem [22]. The verifier approach is related to reward modeling and reranking: rather than asking the model to generate a long explanation, we ask it to assign a scalar quality score to each candidate trace.

The task also connects to recent test-time scaling and verifier research. Cobbe et al. [5] showed that generating multiple candidate solutions and selecting the one ranked highest by a verifier can improve mathematical reasoning. More recent work studies how to allocate test-time computation between sampling and verification, including outcome reward models, process reward models, and majority-vote selection [6]. Process reward models provide step-level feedback, whereas outcome reward models supervise only the final answer; GenPRM further explores generative process reward models as step-level verifiers [7]. Our verifier is closer to an outcome-level verifier because it is trained from final verdict agreement rather than step-level reasoning-quality labels.

3. Task and Data

The task covers English, Spanish, and Arabic. According to the task description, the dataset contains 8,000 English, 2,808 Spanish, and 3,260 Arabic claims, with reasoning paths and evidence for training and validation [1, 4]. The participant output is a ranked score list over candidate reasoning traces, plus

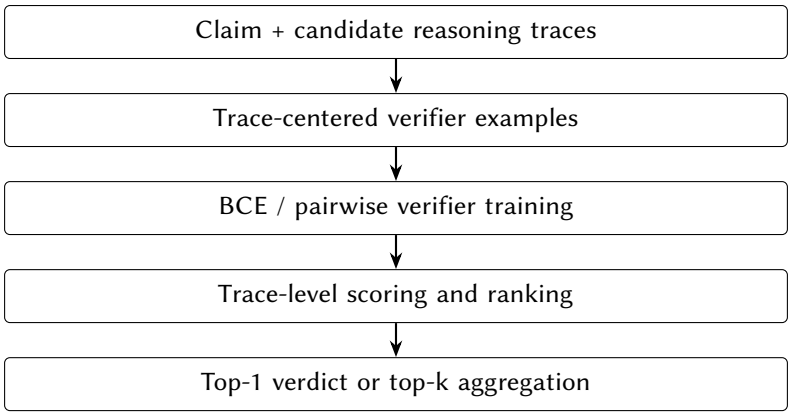

Figure 1: Overview of the verifier-based reasoning-trace ranking pipeline. Training is performed on claim–trace pairs, while evaluation is claim-level ranking and verdict prediction.

the top-ranked verdict. The primary ranking metric is Recall@5; the classification metric is macro-F1 over the veracity labels False, True, and Conflicting. The organizers use both metrics for the final ranking [1].

For the experiments reported here, we used the released train/validation files, the released post-evaluation gold labels, and the official scorer. The reported test scores are offline post-evaluation results rather than official leaderboard entries. We report them to analyze our implementation and to compare model variants under a consistent local evaluation protocol.

The raw test files contain repeated claim texts under different query identifiers, corresponding to multiple evidence settings for the same claim. Following the released scorer, we deduplicate test entries by the `Claim` field before computing the reported scores. No additional test-time augmentation was applied; the only deduplication step in the reported test scores follows the released scorer, which removes repeated claim texts before computing claim-level metrics. After this claim-text deduplication, the local test sets used in our scoring contain 1,708 English claims, 395 Arabic claims, and 1,135 Spanish claims; the raw preprocessing counts are 2,558, 511, and 1,164 entries, respectively. The Arabic test labels do not contain the Conflicting class in our evaluation file; however, we keep the three-label macro-F1 convention for comparability with the scorer.

3.1. Preprocessing and verifier data construction

The original data format is claim-centered: each claim is associated with multiple candidate reasoning traces and candidate verdicts. Our training format is trace-centered: every selected claim–trace pair becomes a binary verifier example. Figure 1 summarizes the conversion and inference pipeline.

For a candidate trace t_i with proposed verdict v_i and gold claim label y, the binary verifier target is

$$z_i = \begin{cases} 1, & \text{if } v_i = y, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

The model input is a concatenation of the claim, the candidate verdict, and the candidate justification. We remove explicit formatting artifacts such as `[Justification]` and `[Label]` markers from the generated traces to reduce superficial leakage from the generator output. This target rewards label agreement rather than explanation faithfulness, so a trace may still receive a positive verifier target if it reaches the correct label for a weak reason.

Verifier input format. Each trace-centered example is serialized as follows during training and inference:

```
Claim: <claim text>
Verdict: <False/True/Conflicting>
Justification: <candidate reasoning trace>
```

Table 1
Preprocessing statistics from the final logs before test-claim deduplication. "Trace len." is the mean number of whitespace tokens per reasoning trace. "Pos. ratio" is the average fraction of candidate trace verdicts matching the gold label and is not applicable at test time.

Lang.	Split	Claims	Traces	Tr./claim	Trace len.	Pos. ratio
EN	train	6,400	127,020	19.8	188.4	0.4796
EN	val	1,600	24,000	15.0	189.1	0.4915
EN	test	2,558	51,160	20.0	110.2	–
AR	train	2,608	52,160	20.0	111.5	0.6944
AR	val	652	13,040	20.0	112.6	0.6975
AR	test	511	10,220	20.0	51.6	–
ES	train	2,246	44,920	20.0	82.1	0.6502
ES	val	562	11,240	20.0	82.2	0.6344
ES	test	1,164	23,280	20.0	89.9	–

Table 2
Raw claim-label distributions and test-time generator verdict distributions before test-claim deduplication. F, T, and C denote False, True, and Conflicting.

Lang.	Train labels	Test labels	Test verdicts
EN	F:3717 T:1175 C:1508	F:1783 T:520 C:255	F:1799 T:488 C:271
AR	F:1444 T:1164	F:318 T:193	F:218 T:261 C:32
ES	F:1888 T:152 C:206	F:837 T:155 C:172	F:821 T:254 C:89

The same textual format is used for encoder and decoder verifiers; only the backbone, pooling strategy, and adaptation method differ across runs.

Table 1 reports claim-level and trace-level preprocessing statistics for all three languages and splits before the scorer deduplicates repeated test claims. Training instances generally have about 20 candidate traces per claim, while the English validation file has 15 traces per claim. English reasoning traces are substantially longer than Arabic and Spanish traces. The average positive trace ratio is also lower for English, with only about 48–49% of English candidate traces matching the gold label in train/validation, compared with about 69–70% for Arabic and 63–65% for Spanish. This makes English the hardest language for the binary verifier target even before model training.

Table 2 gives the corresponding label distributions. English and Spanish include all three classes, whereas the Arabic files used in our local evaluation are binary at the claim-label level even though some candidate traces still contain a Conflicting verdict. This mismatch is one reason why Arabic behaves differently in the top-k and Conflicting-class analyses.

4. System Description

4.1. Models and objectives

We evaluated three model families. First, encoder verifiers use DeBERTa-v3-base, DeBERTa-v3-large, or XLM-RoBERTa-large with a classification head over mean-pooled token states. Second, decoder verifiers use LLaMA-3.2-1B with LoRA adapters. Third, larger decoder verifiers use LLaMA-3.2-3B with 4-bit QLoRA to fit the model on the available local GPU. For decoder models, we tested both mean pooling and a final-position hidden-state pooling variant.

Most runs optimize binary cross-entropy with logits:

$$\mathcal{L}_{bce} = -z \log \sigma(s) - (1 - z) \log(1 - \sigma(s)), \tag{2}$$

where s is the verifier score and z is the binary target. Some variants use a positive-class weight computed from the training data, while others disable this weight to follow the baseline notebook

Table 3
Main training configurations.

Model family	Adaptation	Pooling	Max len.	LR	Epochs
LLaMA-3.2-1B	LoRA	mean/final	256–512	1–2e-4	3
LLaMA-3.2-1B	pairwise LoRA	final	256	1e-4	3
LLaMA-3.2-3B	QLoRA 4-bit	mean	512	2e-4	3 or 5
LLaMA-3.2-3B	pairwise QLoRA	final	256	1e-4	3
XLM-RoBERTa-large	LoRA	mean	512	3e-5	3
DeBERTa-v3	LoRA / full FT	mean	256–512	3e-5–1e-4	3

configuration. We also tried a pairwise ranking loss, inspired by learning-to-rank objectives such as RankNet [16], that directly compares positive and negative traces for the same claim:

$$\mathcal{L}_{pair} = \log\left(1 + \exp(s_{neg} - s_{pos})\right), \tag{3}$$

where s_{pos} and s_{neg} are verifier scores for traces whose verdicts do and do not match the gold label.

4.2. Implementation details

All systems are trained with AdamW, cosine learning-rate scheduling, 5% warmup, and a random seed of 42. Table 3 summarizes the main hyperparameters. We trained and evaluated all systems on one local GPU (NVIDIA GeForce RTX 4070 Ti SUPER). During development we found and fixed an implementation issue in which some inference code read the lowercase key `label`, while the test files used `Label`. All reported test predictions were regenerated after this fix.

5. Experiments

We report validation results for model selection and test results for final analysis. The metrics are Recall@5 for trace ranking, top-1 accuracy for the verdict of the best trace, macro-F1, weighted-F1, and class-wise F1. Unless otherwise noted, boldface in result tables marks the best value within each metric column. For Arabic, the Conflicting class has zero support in the local test labels, so its F1 is not displayed in the Arabic results but still affects the three-label macro-F1. All results are offline scores computed with the released labels and scorer, and should not be interpreted as official leaderboard results.

5.1. Validation results

Table 4 shows a compact English validation comparison. The original LLaMA-3.2-1B baseline from the organizers' notebook was strong. Our LLaMA-3.2-3B QLoRA models produced the best local validation Recall@5 among our own reruns, while XLM-RoBERTa-large was the strongest encoder-only LoRA variant.

5.2. English test results

Table 5 reports English test results. LLaMA-3.2-1B with mean pooling achieved the best Recall@5 and accuracy. DeBERTa-v3-base full fine-tuning achieved the best macro-F1, mainly because its Conflicting F1 was higher than the LLaMA-1B mean-pooling run. This illustrates a trade-off between selecting more correct traces overall and improving minority-class performance.

5.3. Arabic and Spanish test results

Tables 6 and 7 show non-English results. In Arabic, LLaMA-3.2-3B QLoRA trained for five epochs was best by Recall@5, accuracy, and macro-F1. In Spanish, the three-epoch LLaMA-3.2-3B QLoRA run was

Table 4
English validation results for representative systems.

Model	Recall@5	Acc.	Macro-F1	Binary F1
DeBERTa-v3-base LoRA v2	0.2412	0.4719	0.4288	0.7422
XLM-RoBERTa-large LoRA	0.2964	0.5531	0.4564	0.8061
LLaMA-3.2-1B LoRA final	0.2819	0.5481	0.5042	0.8378
LLaMA-3.2-1B full FT	0.2040	0.4088	0.4051	0.8730
LLaMA-3.2-3B QLoRA 3ep	0.3095	0.5800	0.5321	0.8926
LLaMA-3.2-3B QLoRA 5ep	0.3108	0.5794	0.5312	**0.9099**
Original LLaMA-3.2-1B	**0.3123**	**0.5881**	**0.5368**	–

Table 5
English test results on 1,708 claims.

Model	Recall@5	Acc.	Macro-F1	F1-False	F1-True	F1-Conf.
DeBERTa-v3-base full FT	0.2472	0.8009	**0.6030**	**0.8994**	0.6718	0.2378
XLM-RoBERTa-large LoRA	0.2377	0.7845	0.5970	0.8923	0.6492	**0.2494**
LLaMA-3.2-1B LoRA mean	**0.2547**	**0.8162**	0.5722	0.8965	0.6934	0.1268
LLaMA-3.2-1B LoRA final	0.2400	0.7881	0.5828	0.8862	0.6848	0.1774
LLaMA-3.2-3B QLoRA 3ep	0.2512	0.7998	0.5926	0.8943	0.6995	0.1842
LLaMA-3.2-3B QLoRA 5ep	0.2515	0.8015	0.5930	0.8946	**0.7013**	0.1831
LLaMA-3.2-3B pairwise	0.2499	0.7968	0.5600	0.8970	0.6366	0.1463

Table 6
Arabic test results on 395 claims. The Conflicting class is absent in this local test set.

Model	Recall@5	Acc.	Macro-F1	F1-False	F1-True
XLM-RoBERTa-large LoRA	0.2729	0.7620	0.5291	**0.8312**	0.7560
LLaMA-3.2-1B LoRA mean	0.2812	0.7924	0.5344	0.8020	0.8010
LLaMA-3.2-1B LoRA final	0.2813	0.7671	0.5186	0.7700	0.7857
LLaMA-3.2-1B pairwise	0.2695	0.7747	0.5217	0.7801	0.7850
LLaMA-3.2-3B QLoRA 3ep	0.2846	0.7899	0.5380	0.8161	0.7979
LLaMA-3.2-3B QLoRA 5ep	**0.2876**	**0.8000**	**0.5415**	0.8161	**0.8084**
LLaMA-3.2-3B pairwise	0.2738	0.7646	0.5279	0.8080	0.7756

Table 7
Spanish test results on 1,135 claims.

Model	Recall@5	Acc.	Macro-F1	F1-False	F1-True	F1-Conf.
XLM-RoBERTa-large LoRA	0.2091	0.6264	0.5194	0.7510	0.3896	**0.4176**
LLaMA-3.2-1B LoRA mean	0.2586	0.7595	0.5243	0.8702	0.5115	0.1914
LLaMA-3.2-1B LoRA final	0.2631	0.7665	0.6016	0.8739	**0.5485**	0.3826
LLaMA-3.2-1B pairwise	0.2559	0.7559	0.5468	0.8662	0.5178	0.2564
LLaMA-3.2-3B QLoRA 3ep	**0.2727**	**0.7762**	**0.6047**	**0.8776**	0.5263	0.4101
LLaMA-3.2-3B QLoRA 5ep	0.2721	0.7736	0.5973	0.8765	0.5241	0.3911
LLaMA-3.2-3B pairwise	0.2669	0.7665	0.5559	0.8708	0.5233	0.2735

best by Recall@5, accuracy, and macro-F1, while the five-epoch run slightly degraded. These results suggest that the larger QLoRA model benefits multilingual transfer but does not always benefit from longer training.

Across languages, the best Recall@5 systems were LLaMA-1B LoRA mean for English, LLaMA-3.2-3B QLoRA 5ep for Arabic, and LLaMA-3.2-3B QLoRA 3ep for Spanish. We also ran McNemar tests over paired top-1 correctness for selected comparisons. LLaMA-1B LoRA mean significantly improved

Table 8
Summary of the main ablation findings.

Factor	Main evidence	Interpretation
Encoder vs. decoder	Best LLaMA systems beat XLM-R in Recall@5 for all languages.	Decoder pretraining helps score long reasoning traces.
1B vs. 3B decoder	1B is best on English accuracy; 3B QLoRA is best on Arabic and Spanish.	Extra capacity helps more under non-English transfer.
Pooling	Mean pooling is best for English Recall@5; final pooling improves Spanish macro-F1 for 1B.	Pooling affects evidence aggregation and label sensitivity.
BCE vs. pairwise	Pairwise runs are competitive in some Recall@5 cases but lower in macro-F1.	Pure pairwise training does not calibrate final labels well.
Top-1 vs. top-k	EN k=9 and ES k=3 improve macro-F1 but reduce accuracy.	Aggregation helps Conflicting recall but can override correct top traces.

Table 9
Top-k majority-vote aggregation. The default system is k=1. "Conf. F1" is omitted for Arabic because the local Arabic test labels contain no Conflicting examples.

Model and language	Top-1 Acc.	Top-1 MF1	Best k	Best-k Acc.	Best-k MF1	Conf. F1 change
LLaMA-1B LoRA (EN)	0.8162	0.5722	9	0.7986	0.5892	0.1268 → 0.2053
LLaMA-3B QLoRA 5ep (AR)	0.8000	0.5415	1	0.8000	0.5415	N/A
LLaMA-3B QLoRA 3ep (ES)	0.7762	0.6047	3	0.7744	0.6115	0.4101 → 0.4245

English accuracy over both LLaMA-3B QLoRA 5ep and DeBERTa-base full fine-tuning, while LLaMA-3B QLoRA significantly outperformed XLM-RoBERTa-large in Arabic and Spanish; all selected comparisons were significant at $p < 0.05$.

5.4. Compact ablation summary

Table 8 summarizes the main controlled comparisons that guided the final system choice. The table is not intended to replace the full result tables, but to make the model-selection logic explicit. Overall, the best setting depends on the target metric: the English accuracy-oriented choice differs from the English macro-F1 choice, and the best non-English systems favor the larger QLoRA model.

5.5. Top-k verdict aggregation

The official ranking metric rewards placing at least one correct trace among the top five, but our default final verdict uses only the top-ranked trace. We therefore tested majority voting over the verdicts of the top-k traces, breaking ties by the highest individual verifier score among the tied labels. Table 9 summarizes the best top-k result for the best model in each language, and Figure 2 shows the full macro-F1 trend for k in {1,3,5,7,9}. Weighted voting by raw verifier score was tested in our scripts but did not improve over majority voting.

The top-k experiment reveals a trade-off between accuracy and minority-class recall. On English, k=9 improves macro-F1 from 0.5722 to 0.5892, and Conflicting F1 rises from 0.1268 to 0.2053. However, accuracy decreases from 0.8162 to 0.7986. On Spanish, k=3 slightly improves macro-F1 from 0.6047 to 0.6115 and Conflicting F1 from 0.4101 to 0.4245. On Arabic, where the available gold labels contain no Conflicting claims, top-k voting only dilutes the top-ranked decision and decreases both accuracy and macro-F1. For this reason, top-k aggregation is useful as an analysis tool and may be beneficial for macro-F1 optimization, but the top-1 verdict remains the safest default when accuracy is also important.

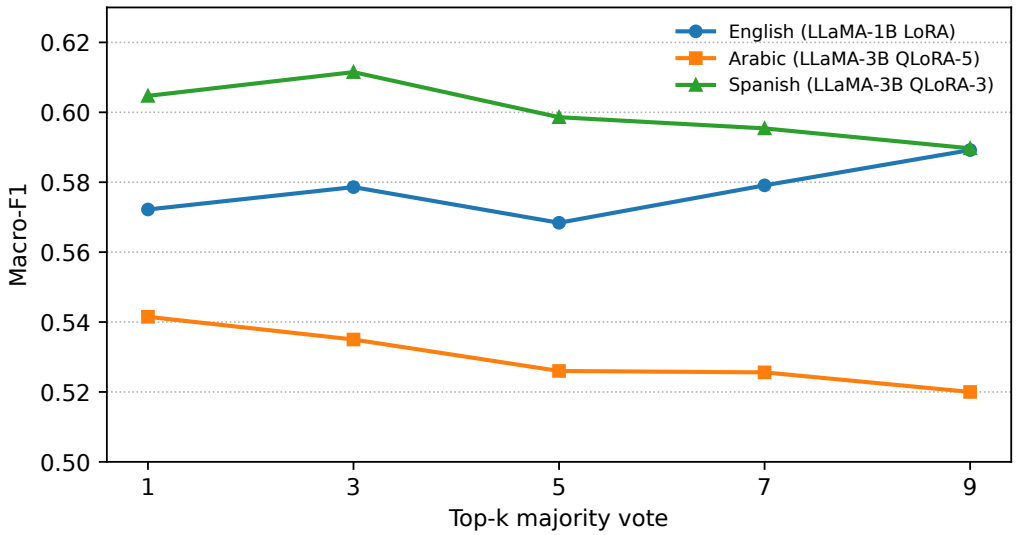

Figure 2: Macro-F1 under top-k majority-vote aggregation for the best local system in each language. English benefits most from larger k, Spanish peaks at k=3, and Arabic degrades as k increases because the local Arabic labels are binary.

6. Analysis

6.1. Model-family and language differences

The best LLaMA-based verifiers outperform encoder LoRA baselines in Recall@5 across all three languages. This is especially visible in Spanish, where LLaMA-3.2-3B QLoRA improves Recall@5 by more than six absolute points over XLM-RoBERTa-large. A likely explanation is that decoder LLM pretraining provides stronger representations for evaluating long reasoning traces, even when only a small classification head and low-rank adapters are trained.

Model size helps more outside English. On English, the smaller LLaMA-3.2-1B mean-pooling model produced the highest Recall@5 and accuracy. On Arabic and Spanish, the 3B QLoRA model was strongest. This suggests that additional capacity is most useful under cross-lingual or non-English transfer, while the smaller model can be sufficient for English when the training distribution is closer to the test distribution. Training longer gives diminishing returns: moving the 3B model from three to five epochs slightly helps Arabic, has negligible effect on English, and slightly hurts Spanish.

Pooling is also not merely an implementation detail. Mean pooling gives the best English accuracy and Recall@5 in our 1B setting, while final-position pooling improves Spanish macro-F1 for LLaMA-3.2-1B. A possible explanation is that final-position pooling is more sensitive to the final verdict and conclusion-like tokens, whereas mean pooling aggregates more evidence from the entire justification.

6.2. Training objective and aggregation

Although the task is ranking-oriented, pure pairwise softplus training underperformed binary cross-entropy in most settings. We suspect that each claim has few positive traces, and the pairwise objective may overfit to local comparisons without learning well-calibrated scores for the final top-1 verdict. Another possible explanation is that Recall@5 does not require only distinguishing the best positive trace from every negative trace; it also benefits from stable global score calibration across claims.

Top-k aggregation partly compensates for this calibration problem. It can recover additional minority-class predictions when multiple high-scoring traces support a label that was not selected at rank one. However, the same mechanism can also override a correct top-ranked trace and therefore reduce accuracy. This explains why k=9 improves English macro-F1 but hurts accuracy, and why top-k voting is harmful in Arabic where the local test labels are binary.

Table 10
Focused Conflicting-class error analysis. “To F” and “To T” count gold Conflicting claims predicted as False or True. “F to C” and “T to C” count False or True claims incorrectly predicted as Conflicting.

Lang.	Model	Conf. acc.	To F	To T	F to C	T to C	N conf.
EN	LLaMA-1B LoRA mean	0.0839	115	27	23	14	155
EN	LLaMA-3B QLoRA 5ep	0.1742	103	25	69	44	155
EN	DeBERTa-base full FT	0.2516	94	22	74	60	155
EN	XLM-RoBERTa LoRA	0.3097	91	16	103	79	155
ES	LLaMA-1B LoRA mean	0.1163	117	35	7	10	172
ES	LLaMA-1B LoRA final	0.3314	87	28	43	26	172
ES	LLaMA-3B QLoRA 3ep	0.3314	92	23	27	22	172
ES	XLM-RoBERTa LoRA	0.5233	60	22	129	40	172

6.3. Conflicting-class errors

Across English and Spanish, F1 for False is high, True is moderate, and Conflicting is substantially lower. In English, the best accuracy model reaches 0.8965 F1 for False and 0.6934 F1 for True, but only 0.1268 for Conflicting. In Spanish, the best macro-F1 model improves Conflicting to 0.4101, but this remains far below the False class. Table 10 shows a focused error analysis.

The main pattern is that all models misclassify many Conflicting claims as False rather than True. The strongest overall English model is also the weakest on Conflicting accuracy, correctly identifying only 8.4% of gold Conflicting claims. XLM-RoBERTa is more balanced and reaches 31.0% Conflicting accuracy in English and 52.3% in Spanish, but it over-predicts Conflicting and therefore sacrifices overall accuracy. This explains why macro-F1 and accuracy sometimes select different models. The Conflicting class likely requires recognizing evidential ambiguity or partial support, which may not be captured by simply checking whether the candidate verdict string matches the training label.

6.4. Data difficulty and calibration

The preprocessing statistics help explain part of the cross-lingual behavior. English has longer reasoning traces and a lower positive trace ratio, making the binary verifier task harder. Arabic has the highest positive trace ratio and no Conflicting examples in the local test labels, which makes top-1 classification easier once the model can identify reliable traces. Spanish is intermediate: its raw positive trace ratio is high, but the Conflicting class is present and non-trivial.

These differences suggest that language-specific calibration may be more useful than a single global threshold or a single aggregation strategy. For example, top-k voting is useful for English and Spanish macro-F1 but harmful for Arabic in our local labels. A future system could tune aggregation and calibration separately per language, or use the distribution of trace scores within a claim as an uncertainty signal.

7. Limitations

The first limitation is evaluation scope: the reported scores are offline post-evaluation scores computed with the released scorer and labels, rather than official leaderboard entries. Thus, they should be interpreted as local ablations and system analysis under a consistent protocol, not as direct leaderboard comparisons. A second limitation is that the verifier target rewards agreement between the candidate verdict and the gold label, not necessarily the logical correctness or faithfulness of the trace. Consequently, two traces that lead to the same correct verdict receive the same positive target even if one provides stronger evidence or more faithful reasoning. This makes our verifier closer to an outcome-level verifier than a process-level verifier. The qualitative Conflicting-class analysis partly exposes this limitation, but future work should incorporate process-level or human-annotated reasoning-quality

signals. A third limitation is that top-k voting was evaluated only as simple post-processing; more principled aggregation could combine calibrated probabilities, trace diversity, and label uncertainty.

8. Conclusion

We presented Leifus, a verifier-based system for CheckThat! 2026 Task 2. The approach ranks generated reasoning traces by fine-tuning parameter-efficient binary verifiers. LLaMA-3.2-1B LoRA was strongest on English Recall@5 and top-1 accuracy, while LLaMA-3.2-3B QLoRA was strongest on Arabic and Spanish. Full fine-tuning and pairwise ranking produced useful comparisons but did not consistently improve the combined objective. Top-k majority voting improved macro-F1 in English and Spanish, mostly by recovering additional Conflicting cases, but also reduced accuracy. The results show that open-source verifier models can be effective for trace selection, while also highlighting the need for better treatment of ambiguous Conflicting cases and more robust calibration of ranking scores.

Reproducibility

The code used for these experiments includes preprocessing scripts, LoRA/QLoRA training scripts, pairwise training, inference, scorer utilities, top-k aggregation, and Conflicting-class analysis. The main random seed is 42. The most important hyperparameters are listed in Table 3. All experiments were run on one local GPU (NVIDIA GeForce RTX 4070 Ti SUPER). All reported scores are computed offline with the released labels and scorer and are not official leaderboard results. We used the official format checker and scorer released by the task organizers for local evaluation.

Declaration on Generative AI

During the preparation of this manuscript, the authors used OpenAI ChatGPT to help draft and organize text from experimental notes, convert result tables into LaTeX, and polish English. The authors reviewed, edited, and verified the final manuscript and remain fully responsible for its content.

References

[1] CheckThat! Lab Organizers. CheckThat! Lab at CLEF 2026: Task 2 Fact-Checking Numerical Claims. https://checkthat.gitlab.io/clef2026/task2/, 2026. Accessed: 2026-05-28.

[2] CheckThat! Lab Organizers. CLEF2026 CheckThat! Lab Task 2 Repository. https://gitlab.com/checkthat_lab/clef2026-checkthat-lab/-/tree/main/task2, 2026. Accessed: 2026-05-28.

[3] Julia Maria Struß, Sebastian Schellhammer, Stefan Dietze, Venktesh V, Vinay Setty, Tanmoy Chakraborty, Preslav Nakov, Avishek Anand, Primakov Chungkham, Salim Hafid, Dhruv Sahnan, and Konstantin Todorov. The CLEF-2026 CheckThat! Lab: Advancing Multilingual Fact-Checking. *arXiv preprint arXiv:2602.09516*, 2026.

[4] Venktesh V, Vinay Setty, Primakov Chungkham, Avishek Anand, and Firoj Alam. Overview of the CLEF-2026 CheckThat! Lab Task 2 on Fact-Checking Numerical Claims. In *CLEF 2026 Working Notes*, CEUR Workshop Proceedings, 2026.

[5] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training Verifiers to Solve Math Word Problems. *arXiv preprint arXiv:2110.14168*, 2021. https://arxiv.org/abs/2110.14168

[6] Charlie Victor Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM Test-Time Compute Optimally Can be More Effective than Scaling Parameters for Reasoning. In *The Thirteenth International Conference on Learning Representations*, 2025. https://openreview.net/forum?id=4FWAwZtd2n

[7] Jian Zhao, Runze Liu, Kaiyan Zhang, Zhimu Zhou, Junqi Gao, Dong Li, Jiafei Lyu, Zhouyi Qian, Biqing Qi, Xiu Li, and Bowen Zhou. GenPRM: Scaling Test-Time Compute of Process Reward Models via Generative Reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, no. 41, pp. 34932–34940, 2026. DOI: 10.1609/aaai.v40i41.40797.

[8] Venktesh V, Abhijit Anand, Avishek Anand, and Vinay 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*, pp. 650–660. Association for Computing Machinery, 2024. DOI: 10.1145/3626772.3657874.

[9] 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)*, pp. 809–819. Association for Computational Linguistics, 2018. DOI: 10.18653/v1/N18-1074.

[10] Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. TabFact: A Large-scale Dataset for Table-based Fact Verification. In *International Conference on Learning Representations*, 2020. https://openreview.net/forum?id=rkeJRhNYDH

[11] Liangming Pan, Xiaobao Wu, Xinyuan Lu, Anh Tuan Luu, William Yang Wang, Min-Yen Kan, and Preslav Nakov. Fact-Checking Complex Claims with Program-Guided Reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6981–7004. Association for Computational Linguistics, 2023. DOI: 10.18653/v1/2023.acl-long.386.

[12] Venktesh V, Vinay Setty, Avishek Anand, Boushra Bendou, Maram Hasanain, Houda Bouamor, Gabriel Iturra-Bocaz, Petra Galuščáková, and Firoj Alam. Overview of the CLEF-2025 CheckThat! Lab Task 3 on Fact-Checking Numerical Claims. In *CLEF 2025 Working Notes*, CEUR Workshop Proceedings, vol. 4038, pp. 709–721. CEUR-WS.org, 2025. https://ceur-ws.org/Vol-4038/paper_53.pdf

[13] Quy Thanh Le, Ismail Badache, Aznam Yacoub, and Maamar El Amine Hamri. LIS at CheckThat! 2025: Multi-Stage Open-Source Large Language Models for Fact-Checking Numerical Claims. In *CLEF 2025 Working Notes*, CEUR Workshop Proceedings, vol. 4038, pp. 1009–1023. CEUR-WS.org, 2025. https://ceur-ws.org/Vol-4038/paper_78.pdf

[14] Mohamed A. Abdallah, Rokayah M. Fekry, and Samhaa R. El-Beltagy. NGU_Research at CheckThat! 2025: An LLM Based Hybrid Fact-Checking Pipeline for Numerical Claims. In *CLEF 2025 Working Notes*, CEUR Workshop Proceedings, vol. 4038, pp. 733–742. CEUR-WS.org, 2025. https://ceur-ws.org/Vol-4038/paper_55.pdf

[15] Anirban Saha Anik, Md Fahimul Kabir Chowdhury, Andrew Wyckoff, and Sagnik Ray Choudhury. ClaimIQ at CheckThat! 2025: Comparing Prompted and Fine-Tuned Language Models for Verifying Numerical Claims. In *CLEF 2025 Working Notes*, CEUR Workshop Proceedings, vol. 4038, pp. 795–803. CEUR-WS.org, 2025. https://ceur-ws.org/Vol-4038/paper_61.pdf

[16] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to Rank using Gradient Descent. In *Proceedings of the 22nd International Conference on Machine Learning*, pp. 89–96. Association for Computing Machinery, 2005. DOI: 10.1145/1102351.1102363.

[17] 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. https://openreview.net/forum?id=nZeVKeeFYf9

[18] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient Finetuning of Quantized LLMs. In *Advances in Neural Information Processing Systems*, vol. 36, 2023. https://proceedings.neurips.cc/paper_files/paper/2023/hash/1feb87871436031bdc0f2beaa62a049b-Abstract-Conference.html

[19] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. DeBERTa: Decoding-Enhanced BERT with Disentangled Attention. In *International Conference on Learning Representations*, 2021. https://openreview.net/forum?id=XPZIaotutsD

[20] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8440–8451. Association for Computational Linguistics, 2020. DOI: 10.18653/v1/2020.acl-main.747.

[21] Meta AI. Llama 3.2 1B and 3B Model Cards. https://huggingface.co/meta-llama/Llama-3.2-1B and https://huggingface.co/meta-llama/Llama-3.2-3B, 2024. Accessed: 2026-05-28.

[22] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45. Association for Computational Linguistics, 2020. DOI: 10.18653/v1/2020.emnlp-demos.6.