A Generative-AI-Driven Claim Retrieval System Capable of Detecting and Retrieving Claims from Social Media Platforms in Multiple Languages

Anonymous ACL submission

Abstract

Online disinformation poses a global challenge, placing significant demands on fact-checkers who must verify claims efficiently to prevent the spread of false information. A major issue in this process is the redundant verification of already fact-checked claims, which increases workload and delays responses to newly emerging claims. This research introduces an approach that retrieves previously fact-checked claims, evaluates their relevance to a given input, and provides supplementary information to support fact-checkers. Our method employs large language models (LLMs) to filter irrelevant fact-checks and generate concise summaries and explanations, enabling fact-checkers to faster assess whether a claim has been verified before. In addition, we evaluate our approach through both automatic and human assessments, where humans interact with the developed tool to review its effectiveness. Our results demonstrate that LLMs are able to filter out many irrelevant fact-checks and, therefore, reduce effort and streamline the fact-checking process.

1 Introduction

The rise of social media has accelerated the spread of false information, posing significant societal, economic and public health risks (Zubiaga et al., 2018). This challenge is further compounded by the multilingual nature of false information, making fact-checking a complex and resource-intensive task for fact-checkers. Fact-checkers often struggle to verify claims across languages, particularly in low-resource settings where limited fact-checking support exists (Hrckova et al., 2024). To address this issue, it is crucial to develop multilingual fact-checking approaches that can assist fact-checkers to identify and verify misinformation efficiently.

One of the key tasks in fact-checking is claim retrieval, also known as previously fact-checked claim retrieval (Pikuliak et al., 2023), where the goal is to identify fact-checks from a database that are the most similar to a given input. This task is crucial, as many claims are not entirely new but rather rephrased or repeated versions of previously debunked misinformation. Efficient retrieval enables fact-checkers to quickly detect repeated claims, reduce redundant efforts, and prioritize emerging or complex claims (Hrckova et al., 2024). However, retrieved results may include fact-checks that are only loosely related or irrelevant, increasing the workload. To mitigate this, LLMs can be leveraged to assess the relevance of retrieved

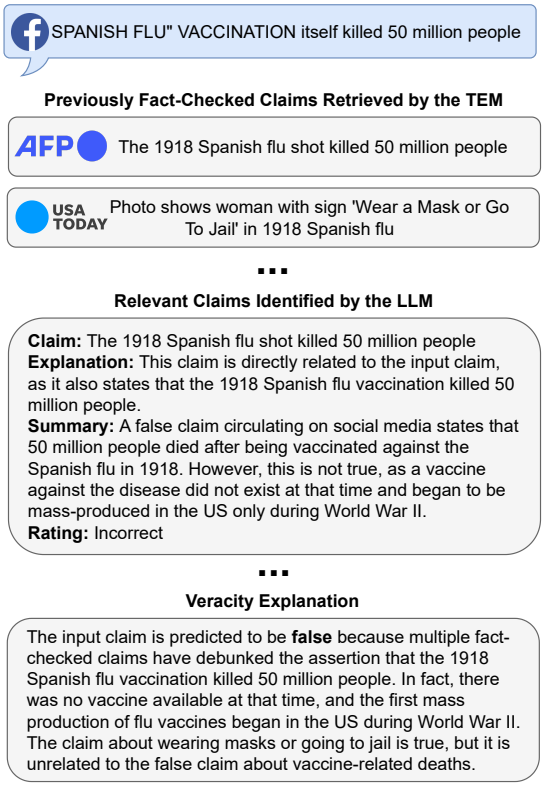

Figure 1: An example of a post with two fact-checked claims retrieved by the embedding model. The LLM selects the relevant claim, explains its choice, summarizes the fact-check article, and predicts the post's veracity.

fact-checks, thereby streamlining the review process (Vykopal et al., 2025).

In this paper, we propose a novel pipeline for retrieving previously fact-checked claims and assisting fact-checkers in assessing their relevance to a given query. Our experiments cover more than 10 languages from diverse linguistic families and scripts, including low- and high-resource languages. We analyze the ability of language models to retrieve relevant fact-checks while incorporating summarization and explanation. An example is shown in Figure 1. In addition, we evaluate LLMs' performance to determine the veracity based on retrieved fact-checks and supporting information.[1]

Our contributions are as follows:

- We provide a novel ***AFP-Sum*** dataset consisting of around 19K fact-checking articles along with their summaries across 23 languages. Additionally, we created a subset of 2300 fact-checks in 23 languages along with the summaries in the original language and translated summaries in English.
- We evaluated multiple text embedding models (TEMs) for retrieving previously fact-checked claims across 20 languages and the capabilities of TEMs for filtering fact-checks based on instructions in the natural language.
- We proposed a novel pipeline for incorporating LLMs into the verification process by employing LLMs for identifying relevant previously fact-checked claims, providing fact-check summaries and predicting the veracity of a given claim based on the previously retrieved fact-checks.

2 Related Work

Previously Fact-Checked Claim Retrieval. Previously fact-checked claim retrieval, also known as verified claim retrieval (Barrón-Cedeño et al., 2020) or claim-matching (Kazemi et al., 2021), aims to reduce fact-checkers' workload by retrieving similar, already verified claims. While most research focused on monolingual settings (Shaar et al., 2020, 2022; Hardalov et al., 2022), multilingual retrieval remains underexplored (Vykopal et al., 2024). Recent work, such as Pikuliak et al. (2023), introduces the ***MultiClaim*** dataset for multilingual claim retrieval, evaluating various TEMs for ranking fact-checked claims in monolingual and cross-lingual contexts.

Recent advancements in LLMs present new opportunities for enhancing claim retrieval. Existing approaches primarily rely on two strategies. The first involves *textual entailment*, where models classify the entailment between an input claim and a fact-check into three categories (Choi and Ferrara, 2024a,b). In contrast, the second strategy employs *generative re-ranking* to rank the previously fact-checked claims based on the conditional probabilities generated by LLMs, which are used to prioritize more relevant claims (Shliselberg and Dori-Hacohen, 2022; Neumann et al., 2023).

Fact-Checking Pipelines & Tools. With the growing importance of online fact-checking, numerous pipelines have been developed to combat misinformation. Many of these systems rely on retrieving the evidence based on a given claim and leveraging LLMs to asses veracity and provide justifications. However, most research has predominantly focused on English (Hassan et al., 2017; Shu et al., 2019; Li et al., 2024) or Arabic (Jaradat et al., 2018; Althabiti et al., 2024; Sheikh Ali et al., 2023), with fewer studies dedicated to other languages.

Several online tools have been developed to address false information. WeVerify[2] provides a suite of tools for identifying false information, including image analysis for detecting manipulated content. In addition, BRENDA (Botnevik et al., 2020) assesses the credibility of claims, helping users evaluate online information. Furthermore, Fact Check Tool[3] aims at retrieving previously fact-checked claims. We build upon the retrieval system and incorporate LLMs into various steps of the entire pipeline to determine the veracity and provide additional information to human fact-checkers.

Multilingual Summarization. Multilingual summarization has been propelled by the development of extensive datasets and the application of LLMs (Scialom et al., 2020; Hasan et al., 2021; Bhattacharjee et al., 2023). These resources have enabled the fine-tuning of multilingual models like mT5 (Xue et al., 2021), which demonstrate competitive performance in both multilingual and low-resource summarization tasks. Furthermore,

[1]The data are available at Zenodo upon request *for research purposes only*: anonymous. The source code is available at: https://anonymous.4open.science/r/claim-retrieval-0925.

[2]https://weverify.eu/

[3]https://toolbox.google.com/factcheck/explorer/

studies have explored the zero-shot and few-shot capabilities of LLMs such as GPT-3.5 and GPT-4 in cross-lingual summarization, highlighting their potential to handle diverse language pairs without extensive fine-tuning (Wang et al., 2023). Efforts to enhance factual consistency in multilingual summarization have also been made, exemplified by the use of multilingual models to improve the reliability of machine-generated summaries across languages (Aharoni et al., 2023).

3 Methodology

Our experiments aim to evaluate the capabilities of TEMs and LLMs in assisting fact-checkers by providing additional information. This includes retrieving the most similar previously fact-checked claims, summarizing fact-checking articles along with their ratings, and potentially predicting the veracity of a given input based on the retrieved information. Much of this process can be automated using LLMs, thereby reducing the effort required from fact-checkers to identify relevant fact-checks.

Our proposed pipeline, illustrated in Figure 2, consists of four key steps: *retrieval* (Section 4), *filtration* (Section 5), *summarization* (Section 6) and *veracity prediction* (Section 7). In the *retrieval* step, the TEM retrieves the top K most similar fact-checks based on a given input. The *filtration* step then refines these results by using the LLM to identify only the fact-checks that are directly relevant, providing explanations for its selection and filtering out irrelevant claims. In the *summarization* step, the LLM generates concise summaries of the relevant fact-checking articles. Finally, in the *veracity prediction* step, the LLM leverages the retrieved fact-checks, their ratings, and the generated summaries to assess and predict the veracity of the given input based on the available information.

In addition, we provide an overview of the datasets (Section 3.1) and models (Section 3.2) used in our experiments. We also detailed the evaluation for each step of the pipeline in Section 3.3.

3.1 Datasets

MultiClaim. We selected the ***MultiClaim*** dataset (Pikuliak et al., 2023) to evaluate the efficiency of embedding models and LLMs in retrieving previously fact-checked claims and assessing claim veracity. ***MultiClaim*** comprises 206K fact-checking articles in 39 languages and 28K social media posts in 27 languages. Additionally, this dataset includes 31K pairs between social media posts and fact-checking articles, linking posts to corresponding fact-checking articles. Moreover, each fact-checking article is assigned a veracity rating and contains a URL, allowing retrieval of the full article content. This supplementary information enhances our pipeline by enabling a more structured and comprehensive evaluation of detecting previously fact-checked claims.

AFP-Sum. To evaluate the abilities of LLMs to summarize fact-checking articles, we created the ***AFP-Sum*** dataset, consisting of fact-checking articles and their summaries from the AFP (*Agence France-Presse*)[4]. We scrapped fact-checking articles across 23 languages, yielding approximately 19K fact-checking articles with summaries written by fact-checkers. For our experiments, we selected 100 fact-checking articles per language, evaluating LLM-generated summaries in English. To facilitate evaluation, we employed Google Translate to translate all reference summaries into English. Table 8 in Appendix B.2 presents the dataset statistics.

3.2 Language Models

We employed two categories of models, especially *text embedding models* and *large language models*. TEMs were used in the retrieval stage to identify the most relevant fact-checks for a given input. While numerous TEMs exist, we selected both English and multilingual models, using BM25 as a baseline for comparison. The TEMs used in our study are listed in Table 2.

In addition to TEMs, we evaluated a diverse set of LLMs, including both closed and open-sourced, chosen based on their state-of-the-art performance across NLP tasks. For the summarization, we also experimented with smaller LLMs with fewer than 3 billion parameters. The full list of LLMs used in our experiments is shown Table 1.

3.3 Evaluation

We employed various evaluation metrics tailored to different stages of our proposed pipeline.

In the retrieval step, we used *success-at-K* (S@K) as the primary evaluation metric for assessing TEMs performance. S@K measures the percentage of cases where a correct fact-check appears within the top K retrieved results. Additionally, we apply this metric to evaluate the ability of LLMs to

[4] https://www.afp.com

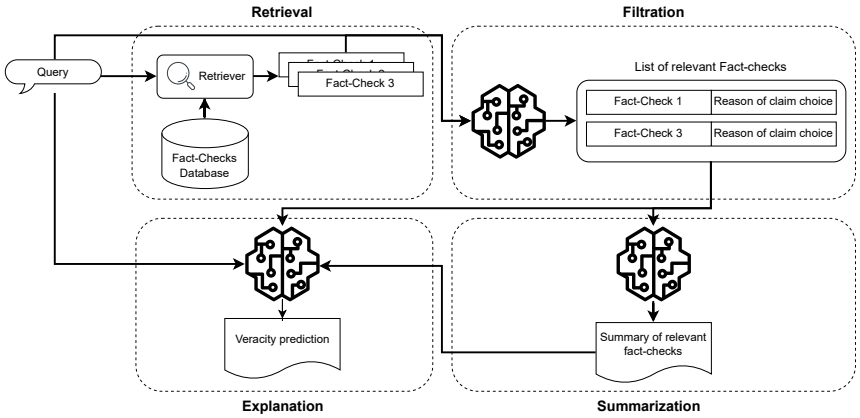

Figure 2: Our proposed pipeline consisting of (1) retrieval of the top N most similar fact-checks, (2) identifying relevant fact-checked claims, (3) summarizing relevant fact-checking articles, and (4) predicting the veracity of the query along with the explanation.

Model	# Params [B]	# Langs	Organization	Citation
`GPT-4o (2024-08-06)`	N/A	N/A	OpenAI	
`Claude 3.5 Sonnet`	N/A	N/A	Anthropic	
`Mistral Large`	123	11	Mistral AI	Mistral AI Team (2024)
`C4AI Command R+`	104	23	Cohere For AI	Cohere For AI (2024)
`Qwen2 Instruct`	72	29	Alibaba	Yang et al. (2024)
`Qwen2.5 Instruct`	0.5, 1.5, 3, 72	29	Alibaba	Yang et al. (2024)
`Llama3.1 Instruct`	70	8	Meta	Grattafiori et al. (2024)
`Llama3.2 Instruct`	1, 3	8	Meta	Grattafiori et al. (2024)
`Llama3.3 Instruct`	70	8	Meta	Grattafiori et al. (2024)
`Gemma3`	27	140	Google	Team et al. (2025)

Table 1: LLMs used in our experiments, including both closed-source and open-source models.

identify the most relevant fact-checks from the set retrieved by a TEM.

For summarization experiments, we used two standard metrics: *BERTScore* and *ROUGE-L*. BERTScore evaluates semantic similarity by computing the F1 score based on contextual word embeddings from a BERT model. ROUGE, on the other hand, measures n-gram overlaps between the generated summary and the reference summary. Specifically, we employed ROUGE-L, which focuses on the longest common subsequence of words. ROUGE-L also helps detect cases where the LLM generated summaries in a language other than English – something that is more challenging to identify using BERTScore.

Finally, for veracity prediction experiments, we employed standard classification metrics for imbalanced data: Macro F1 score, Precision and Recall.

4 Retrieval Experiments

In this section, we describe experiments with various TEMs in two settings. First, *direct retrieval* (Section 4.1), in which we evaluate the performance of existing TEMs for retrieving the most similar previously fact-checked claims based on posts. Second, *criteria-based retrieval* (Section 4.2), where we evaluate TEMs for filtering the results based on criteria specified in a natural language in English (e.g. the presence of a specific named entity, the publication date, etc.).

4.1 Direct Retrieval

We evaluated various TEMs and their performance for ranking previously fact-checked claims based on a given input. We formulate the task as a ranking problem, where we aim to rank all fact-checks from the database for a given input based on cosine similarity. We selected 20 languages with at least 100 posts per language with a setup similar to that proposed by Pikuliak et al. (2023). In addition to the TEMs evaluated in (Pikuliak et al., 2023), we included more recent multilingual TEMs, especially multilingual E5 models of various sizes. We evaluate the TEM's performance only in a monolingual setting, where fact-checked claims and posts are in the same language.

Results. The results of TEMs in the monolingual setting are shown in Table 2. These results demonstrated that **some multilingual TEMs can outperform the combination of English translation with English TEMs**, but not statistically significantly. `Multilingual E5 Large` achieved the highest S@10 (statistically significant; $p < 0.05$), while `GTR-T5-Large` achieved the best results with English translations ($p < 0.05$). The other multilingual TEMs fall behind the English TEMs.

Table 9 shows the results of all studied TEMs across 20 languages. Based on the results of the

Model	Size [M]	Ver.	Avg. S@10
BM25		Og	0.62
English TEMs			
`DistilRoBERTa`	82	En	0.75
`MiniLM-L6`	22	En	0.79
`MiniLM-L12`	33	En	0.79
`MPNet-Base`	109	En	0.77
`GTE-Large-En`	434	En	0.80
`GTR-T5-Large`	737	En	**0.83**
Multilingual TEMs			
`BGE-M3`	568	Og	0.82
`DistilUSE-Base-Multilingual`	134	Og	0.66
`LaBSE`	470	Og	0.69
`Multilingual E5 Small`	117	Og	0.78
`Multilingual E5 Base`	278	Og	0.78
`Multilingual E5 Large`	559	Og	**0.84**
`MiniLM-L12-Multilingual`	117	Og	0.63
`MPNet-Base-Multilingual`	278	Og	0.69

Table 2: Average performance of English and multilingual TEMs in monolingual settings using S@10 metric. Ver. denotes either *original* (Og) or the *English* (En) version of the dataset. The best results for both versions are highlighted in **bold**.

English TEMs, `GTR-T5-Large` achieved superior performance for most languages ($p < 0.05$). However, for the German language, the results were lower than 0.70. On the other hand, `Multilingual E5 Large` proved to be effective across all languages ($p < 0.05$), except Thai, where the smaller `Multilingual E5` outperformed larger versions.

4.2 Criteria-based Retrieval

In addition to retrieval based only on the input claim or posts, we also experimented with criteria-based retrieval, where we employ TEMs to filter the results based on given criteria, e.g., the presence of a specific named entity. The aim is to evaluate whether TEMs can be employed to filter the results with natural language instructions provided by fact-checkers. We defined four settings for the experiments: filtering based on the *language*, *date range*, *fact-checking domain* or the *named entity*. We selected the best-performing TEM – `Multilingual E5 Large` for the experiments. We proposed a template illustrated in Figure 5, consisting of fact-checked claims and metadata, such as language, fact-checking organization, and publish date.

As ground truth, we used the results obtained by using `Multilingual E5 Large` to rank a subset of the data already filtered based on a given condition using the manually-designed filter (e.g., only Spanish fact-checks were ranked). Our pipeline for criteria-based retrieval consists of two steps:

Settings	Avg. Spearman	Avg. Kendall's Tau	Avg. Common FCs
Named Entities	-0.31	-0.20	0.32
Languages	-0.58	-0.43	0.17
Domains	-0.66	-0.51	0.12
Dates	-0.82	-0.64	0.02

Table 3: Scores for average *Spearman correlation coefficient*, *Kendall's Tau* and the *proportion of the common fact-checks* (FCs) between the ground truth and predicted ranked list. We report the mean score across all settings with at least 100 fact-checks per category.

(1) Retrieval based on the criteria (e.g., a given language), where we select only fact-checks with a similarity score of more than 0.8; *(2) Ranking based on the post*, where we rank previously retrieved results using the post content, similarly to the direct retrieval.

For the evaluation, we employed *Spearman's rank correlation coefficient* and *Kendall's Tau* to evaluate the capabilities of TEMs to rank the results by using queries in the natural language. We calculated the correlation between ranks produced by `Multilingual E5 Large` with our two-step approach and ranks obtained by using the `Multilingual E5 Large` on the already filtered results based on manually-designed filters.

Results. Table 3 presents the results for the filtered retrieval across four settings: named entities, languages, fact-checking domains and date ranges. We calculated the average Spearman's rank correlation, Kendall's Tau and the proportion of the common fact-checks between the predicted and reference list of fact-checks. A positive correlation indicates that the predicted ranking aligns with the reference ranking, whereas a negative correlation suggests an inverse relationship.

Our results showed that **filtering based on the named entities yielded the highest overlap between the predicted and ground truth fact-check lists** ($p < 0.05$), suggesting that TEMs performed best when fact-checks were retrieved based on named entities. Despite this, the Spearman correlation of -0.31 indicates that while TEMS might identify relevant fact-checks, their rankings did not fully match the ground truth ordering.

Filtering by language, domain and date range led to lower performance, with the latter performing the worst. This suggests that while TEMs can retrieve relevant fact-checks based on natural language instructions, their filtering changes the can-

didate set that limits and reduces overall ranking performance. Additionally, we specified a date range, whereas the embeddings of fact-checks only included the exact date of each fact-check. This discrepancy made it more challenging for TEMs to retrieve the fact-checks based on dates not explicitly mentioned in the prompt.

5 Filtration Experiments

To filter out irrelevant previously fact-checked claims, we experimented with several LLMs on a subset of the ***MultiClaim*** dataset. We selected 10 languages, specifically *Czech*, *English*, *French*, *German*, *Hindi*, *Hungarian*, *Polish*, *Portuguese*, *Spanish* and *Slovak*, with 100 posts per language. These posts were chosen based on their veracity labels. However, since the ***MultiClaim*** dataset predominantly contains false posts, achieving a balanced distribution was not feasible. The final dataset consists of 55 true, 65 unverifiable and 880 false posts, resulting in a significant imbalance. We used this data to evaluate the efficiency of LLMs in filtering out irrelevant fact-checks for a given input.

Our approach involved a two-step process. First, we used `Multilingual E5 Large` to retrieve the 50 most similar fact-checked claims. Then, we instructed the LLM to filter this set (see Figure 7), selecting only those directly relevant to the input post, while removing irrelevant fact-checks.

To assess the performance and efficiency of the LLMs in this task, we calculated the S@10 and MRR (mean reciprocal rank) scores for retrieval. In addition, we calculated Macro F1, True negative rate (TNR) and False negative rate (FNR) to identify the capabilities of LLMs. To calculate classification metrics, we created pairs of posts and fact-checks identified by the `Multilingual E5 Large` model, where the relevance labels were obtained from the labelled pairs from the ***MultiClaim*** dataset. TNR represents the proportion of how many irrelevant fact-checks were correctly filtered out, while FNR represents the proportion of how many relevant fact-checks were incorrectly filtered out. In this case, we want to maximize the TNR and minimize the FNR.

5.1 Results

Table 4 summarizes our results on filtering irrelevant fact-checks. Using `Multilingual E5 Large` as the baseline, we correctly retrieved 76% of relevant fact-checks within the top 10 results. To further assess performance, we framed the ranking task as binary classification, selecting an optimal threshold using Youden's Index. Macro F1 showed that the baseline outperformed `Llama3.1 8B`.

Model	S@10 ↑	MRR ↑	Macro F1 ↑	TNR ↑	FNR ↓
`Multilingual E5 Large`	**0.76**	**0.58**	54.75	86.27	25.59
`Mistral Large 123B`	<u>0.70</u>	<u>0.40</u>	<u>59.82</u>	90.23	**15.38**
`C4AI Command R+`	0.66	0.35	55.50	85.83	**15.38**
`Qwen2.5 72B`	0.57	0.32	58.37	90.81	30.65
`Llama3.3 70B`	0.67	0.38	**59.96**	90.82	<u>19.61</u>
`Llama3.1 70B`	0.63	0.37	59.62	<u>91.08</u>	24.25
`Gemma3 27B`	0.65	0.35	57.77	89.14	21.78
`Llama3.1 8B`	0.60	0.24	52.38	82.30	21.16
`Qwen2.5 7B`	0.47	0.35	59.25	**93.20**	43.86

Table 4: Retrieval and filtration performance results on 100 posts across 10 languages. `Multilingual E5 Large` serves as the baseline. The best results are highlighted in **bold**, while the second-best results are <u>underlined</u>.

After retrieval, we applied an LLM to filter the top 50 retrieved fact-checks. While this lowered S@10 and MRR scores compared to the baseline, the aim was to reduce the number of irrelevant fact-checks presented to fact-checkers. We measured the proportion of relevant and irrelevant fact-checks removed. **`Mistral Large` achieved the best trade-off between TNR and FNR** ($p < 0.05$), while also outperforming other LLMs in S@10.

Our findings suggest that **while LLMs effectively remove irrelevant fact-checks, they may exclude some relevant ones**. The performance gap between `Multilingual E5 Large` and LLMs indicates occasional misclassification of relevant fact-checks as irrelevant, although LLMs may also elevate lower-ranked fact-checks into the top 10.

6 Summarization Evaluation

We evaluated LLMs on summarizing fact-checking articles using a subset of our ***AFP-Sum*** dataset across 23 languages. Experiments were conducted in two settings: (1) *Article first* – the article is provided before the instruction; (2) *Article last* – the article is provided after the instruction (see Figure 6). We examined how prompt order and quantization affect summary quality. Articles were provided in their original language, with instructions to generate a summary in English. The generated summaries were compared against English translations of reference summaries using Google Translate.

6.1 Results

Figure 3 presents the overall results using the ROUGE-L metric in the *Article first* setup. The re-

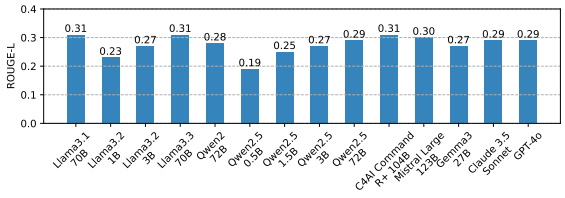

Figure 3: Overall performance of LLMs for fact-check summarization. We report the average ROUGE-L score for each LLM using the *Article first* setup, where the article is provided before the instruction.

sults demonstrated the diverse performance across LLMs. **Smaller `Llama` models struggled with summarization, often generating summaries in the article's original language** instead of English, leading to lower ROUGE-L. In contrast, other LLMs better adhered to the instruction to produce English summaries. Furthermore, providing the instruction before the article worsened this issue and resulted in a very low ROUGE-L (see Table 13).

Table 12 compares `Llama3.2` models (1B and 3B parameters) across different article-order setups and three quantization levels. Summaries were generated using 4-bit, 8-bit, and full-precision models. Results showed that **providing the article before the instruction significantly improved performance** ($p < 0.05$), yielding better results when the article was provided after the instruction. For `Llama3.2 1B`, the 8-bit model generally performed best (not statistically significant), with full precision close behind. The performance gap between the full-precision and 4-bit was around 0.3 BERTScore points for `Llama3.2 1B`, while for `Llama3.2 3B`, only 0.1, suggesting that **quantization has less impact on larger LLMs**.

Overall, `Llama3.3 70B` and `Mistral Large` achieved the best performance across languages (see Table 11), while other LLMs lagged behind (not statistically significant). The results indicate that LLMs covering fewer languages (e.g., `Llama`) can outperform broader multilingual LLMs, e.g., `C4AI Command R+` or `Qwen2.5`.

7 Evaluation of LLM's Veracity Prediction

To assess how well LLMs predict claim veracity using retrieved previously fact-checked claims and the fact-check summaries (see Figure 7), we employed the same data as in Section 5. The final dataset consists of three classes: *True*, *False* or *Unverifiable*, which are imbalanced. Therefore, we leveraged Macro F1, Macro Precision and Macro Recall to evaluate the performance of LLMs. In this case, the supplementary information is in English, particularly summaries, ratings and fact-checked claims. This is also beneficial for human fact-checkers to understand the results provided by LLMs. As baselines, we selected `Mistral Large` and `Llama3.3`, instructed only with the post and task description without additional information.

Model	Macro F1	Macro Precision	Macro Recall
Baseline (without retrieved fact-checks)			
`Mistral Large 123B`	26.53	39.57	33.25
`Llama3.3 70B`	30.29	34.22	33.30
`Mistral Large 123B`	**63.05**	**64.88**	**61.62**
`C4AI Command R+`	54.92	55.50	54.38
`Qwen2.5 72B`	57.28	57.28	57.33
`Llama3.3 70B`	52.62	52.18	53.09
`Llama3.1 70B`	51.68	50.67	53.25
`Gemma3 27B`	52.39	52.62	52.36
`Llama3.1 8B`	49.15	46.66	53.65
`Qwen2.5 7B`	51.99	56.47	49.23

Table 5: Veracity prediction results across various LLMs. Results are presented for baseline (without retrieved fact-checks), and LLMs with retrieved fact-checks. The best results are highlighted in **bold**.

7.1 Results

Our results are shown in Table 5, where we employed eight LLMs with different model sizes. The **`Mistral Large` with the retrieved information outperformed all other LLMs**, also the baselines (not statistically significant). It achieved the highest performance, making it the most reliable for veracity prediction out of the experimented LLMs.

`Qwen2.5 72B` follows with a noticeable drop in performance, suggesting that model size alone does not determine effectiveness. `Llama` models performed similarly, showing limited ability to distinguish veracity class based on the retrieved information. **The smaller models performed the worst and struggled with generalization.**

Overall, while bigger LLMs tend to perform better, **the contextual information plays an important role**. The strong performance of `Mistral Large` highlights its potential for improving fact-checking applications.

8 Human Evaluation

To assess the effectiveness of our proposed pipeline for multilingual claim retrieval, we developed a

web-based tool designed for fact-checkers. In addition to conducting automatic evaluations of individual components, we focused on human evaluation of the entire pipeline using the developed tool. We provided the tool to students and academics, who assessed its performance and usability. Their feedback was collected through the evaluation workshop and a structured questionnaire, offering insights into the system's applicability.

8.1 Developed Tool

Our web-based application integrates the pipeline described in Section 3, utilizing the best-performing TEM model – `Multilingual E5 Large`. The backend employs `Llama3.3 70B`, selected for its strong summarization ability and ability to filter irrelevant fact-checks. We store fact-checked claims from over 80 languages, along with metadata and `Multilingual E5 Large` embeddings, in the Milvus[5] vector database. The result of the system provides a ranked list of relevant fact-checks identified by the LLM, along with their summaries and explanations. In addition, we provide users with a veracity label distribution graph and a verdict explanation to aid decision-making.

8.2 Evaluation & Results

To evaluate the tool, we conducted a user study with six participants (five journalism students and one academic). Each participant interacted with the tool and completed the questionnaire designed to assess system usability, output quality and overall effectiveness in supporting fact-checking.

Participants rated their satisfaction with various aspects of the tool, including summaries, explanations of relevant fact-checks, the veracity graph and overall usability. Ratings ranged from 1 (very unsatisfied) to 5 (very satisfied). Most features received average satisfaction scores between 3.5 and 4, with explanations of relevant fact-checks and veracity explanations achieving the highest average rating. Users generally found the tool helpful in identifying relevant fact-checks and appreciated the clarity of summaries and explanations.

We asked participants to assess the tool's main benefits (see Figure 4). Participants highlighted the retrieval of relevant fact-checks (FCs), concise summaries and explanations, and the clarity of the interface as the main benefits. These results suggest that the tool is a promising aid for fact-checkers.

[5] https://github.com/milvus-io/milvus

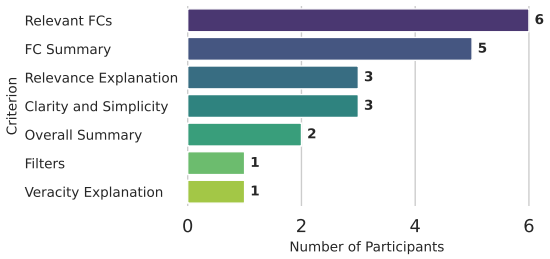

Figure 4: Number of participants ($N = 6$) who highlighted each evaluation criterion as beneficial.

9 Discussion

Multilingual TEMs Outperform English TEMs. `Multilingual E5 Large` achieved the best retrieval performance across most languages. However, **criteria-based retrieval** experiments showed that TEMs struggled with natural language instructions, especially when filtering by date range.

Filtration with LLMs Improves Precision But Has Trade-Offs. `Mistral Large` provided the best balance between retaining relevant fact-checks and filtering out irrelevant ones, showing promise for assisting fact-checkers. However, the trade-off between precision and recall remains challenging, as some useful fact-checks may be excluded.

Larger LLMs Excel in Summarization and Veracity Prediction. Smaller LLMs often failed to follow instructions, producing summaries in the original language instead of English. Larger LLMs performed better, particularly when the article preceded the instruction, and were more effective at predicting claim veracity. However, overall performance remained moderate due to the inherent difficulty of accurately assessing claim veracity.

10 Conclusion

This paper presents a pipeline for multilingual retrieval of previously fact-checked claims, integrating LLMs to enhance the fact-checking process. Beyond the retrieval, our approach supports fact-checkers by filtering irrelevant fact-checks, summarizing fact-checking articles, and predicting veracity labels along with explanations. We also developed a web-based application and evaluated its effectiveness in the fact-checking process. Our findings demonstrate the potential of LLMs to improve fact-checking workflows, making them more efficient and accessible across languages.

Limitation

Models Used. Our experiments relied on a selection of state-of-the-art LLMs and TEMs, including closed-source models (e.g., `GPT-4o` or `Claude 3.5 Sonnet`) and open-sourced models (e.g., `Mistral Large`, `Llama3`). However, model performance is highly dependent on training data and fine-tuning strategies. As a result, our findings may not generalize to all LLMs and architectures, and future improvements may arise with newer models.

Language Support. Despite evaluating our approach on more than 10 languages and incorporating fact-checking data from 20 languages, our system may still face challenges in handling low-resource languages. The performance of TEMs and LLMs may vary across languages, particularly those with limited pre-trained resources.

In addition, the selected LLMs exhibit varying degrees of multilingual capabilities. While model cards on Hugging Face[6] indicate intended language support, the models may demonstrate capabilities in additional languages due to the training data diversity and potential data contamination.

Human Evaluation. Our user study included six participants from an academic environment – five journalism students and one academic. While professional fact-checkers would have been more appropriate evaluators for our tool, their inclusion was not feasible due to time constraints and limited availability. Journalism students, however, serve as a reasonable proxy, given their specialization and relevance as potential end-users. We acknowledge this limitation and consider evaluation with professional fact-checkers as an important direction for future work.

Automated Veracity Prediction. Our pipeline includes an LLM-based veracity prediction, which suggests a claim's veracity based on retrieved fact-checks. However, automated assessments remain limited by the availability and accuracy of fact-checking data. In cases where no relevant fact-check exists, the system may struggle to provide reliable predictions.

Ethical Consideration

Biases. Since we experimented with LLMs, our system may inherit biases from the training data used in the embedding models and LLMs. These biases can affect claim retrieval, relevant fact-check selection, and veracity assessments, potentially leading to skewed or misleading outputs, especially for politically sensitive or controversial topics.

Additional bias is propagated by fact-checkers since they decide what they will fact-check.

Developed Tool. The final version of the developed tool employs the `Llama3.3 70B`[7]. This model was selected for its advanced capabilities in summarizing and efficient inference, compared to larger models. The tool includes biases inherited from the used LLM.

To enhance transparency and assist users in evaluating the output, the tool also provides the number of supporting and refuting fact-checks associated with a given claim. This information is included in the veracity distribution graph within the tool. The user can employ this information for the final decision on the veracity of the given claim and compare the veracity prediction done by the LLM.

The classification accuracy and efficiency of the pipeline depend on the final model – in our case, `Llama3.3 70B`. The evaluation of the corresponding model and its effectiveness for the veracity prediction is evaluated in Section 7. LLMs are known to hallucinate (Rawte et al., 2023), and therefore, they might create fake, non-factual or even harmful information.

In addition, the tool incorporates fact-checking articles and corresponding claims, many of which are false or misleading statements spread online. As a result, users of the application may be exposed to false, misleading, or even harmful claims. To address this, the tool includes a Terms of Use that outlines its intended purpose, identifies the target users, and specifies user groups for whom the tool is not intended.

Personal Information. The original ***MultiClaim*** (Pikuliak et al., 2023) dataset might contain personal information and data from the social media posts (e.g., the names of users). However, we are not using any personal information within our experiments or the developed tool.

Terms of Use of Platforms and Datasets. In our research, we utilized the ***MultiClaim*** dataset (Pikuliak et al., 2023), which is accessible under specific conditions – the dataset is restricted to academic and research purposes.

[6] https://huggingface.co/

[7] https://huggingface.co/meta-llama/Llama-3.3-70B-Instruct

Additionally, we incorporated fact-checking articles from Agence France-Presse (AFP)[8], which are available for personal, private, and non-commercial use. Any reproduction or redistribution beyond these permitted uses is forbidden.

We ensure that our use of both the ***MultiClaim*** and AFP's content for the ***AFP-Sum*** is compiled with their respective terms and conditions.

Intended Use. The annotated data presented in this research are intended solely for research purposes. They are derived from the existing ***MultiClaim*** dataset (Pikuliak et al., 2023), which is also intended only for research purposes. In our work, we selected a subset and annotated specific portions for the task of veracity prediction.

Additionally, we introduce the ***AFP-Sum*** dataset, comprising fact-checking articles and their summaries sourced from the AFP organization. Due to the copyright restrictions on the AFP data, its usage is strictly limited to research purposes. As such, we release the ***AFP-Sum*** dataset and any derived resources to researchers for non-commercial research use only.

To promote reproducibility, we also release code used to obtain our results. Both the datasets and the code are intended only for research use, and replicating our findings requires access to the original ***MultiClaim*** dataset, which is available under its respective terms and conditions.

Usage of AI assistants. We have used the AI assistant for grammar checks and sentence structure improvements. We have not used AI assistants in the research process beyond the experiments detailed in the Methodology section (Sec. 3).

References

Roee Aharoni, Shashi Narayan, Joshua Maynez, Jonathan Herzig, Elizabeth Clark, and Mirella Lapata. 2023. Multilingual summarization with factual consistency evaluation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3562–3591, Toronto, Canada. Association for Computational Linguistics.

Saud Althabiti, Mohammad Ammar Alsalka, and Eric Atwell. 2024. Ta'keed: The first generative fact-checking system for arabic claims. *Preprint*, arXiv:2401.14067.

Alberto Barrón-Cedeño, Tamer Elsayed, Preslav Nakov, Giovanni Da San Martino, Maram Hasanain, Reem Suwaileh, Fatima Haouari, Nikolay Babulkov, Bayan Hamdan, Alex Nikolov, Shaden Shaar, and Zien Sheikh Ali. 2020. Overview of checkthat! 2020: Automatic identification and verification of claims in social media. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 215–236, Cham. Springer International Publishing.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Uddin Ahmad, Yuan-Fang Li, Yong-Bin Kang, and Rifat Shahriyar. 2023. CrossSum: Beyond English-centric cross-lingual summarization for 1,500+ language pairs. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2541–2564, Toronto, Canada. Association for Computational Linguistics.

Bjarte Botnevik, Eirik Sakariassen, and Vinay Setty. 2020. Brenda: Browser extension for fake news detection. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 2117–2120, New York, NY, USA. Association for Computing Machinery.

Eun Cheol Choi and Emilio Ferrara. 2024a. Automated claim matching with large language models: Empowering fact-checkers in the fight against misinformation. In *Companion Proceedings of the ACM Web Conference 2024*, WWW '24, page 1441–1449, New York, NY, USA. Association for Computing Machinery.

Eun Cheol Choi and Emilio Ferrara. 2024b. Fact-gpt: Fact-checking augmentation via claim matching with llms. In *Companion Proceedings of the ACM Web Conference 2024*, WWW '24, page 883–886, New York, NY, USA. Association for Computing Machinery.

Cohere For AI. 2024. c4ai-command-r-plus-08-2024 (revision dfda5ab).

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Momchil Hardalov, Anton Chernyavskiy, Ivan Koychev, Dmitry Ilvovsky, and Preslav Nakov. 2022. CrowdChecked: Detecting previously fact-checked claims in social media. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 266–285, Online only. Association for Computational Linguistics.

Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang,

[8] https://factcheck.afp.com/

M. Sohel Rahman, and Rifat Shahriyar. 2021. XL-sum: Large-scale multilingual abstractive summarization for 44 languages. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4693–4703, Online. Association for Computational Linguistics.

Naeemul Hassan, Gensheng Zhang, Fatma Arslan, Josue Caraballo, Damian Jimenez, Siddhant Gawsane, Shohedul Hasan, Minumol Joseph, Aaditya Kulkarni, Anil Kumar Nayak, Vikas Sable, Chengkai Li, and Mark Tremayne. 2017. Claimbuster: the first-ever end-to-end fact-checking system. *Proc. VLDB Endow.*, 10(12):1945–1948.

Andrea Hrckova, Robert Moro, Ivan Srba, Jakub Simko, and Maria Bielikova. 2024. Autonomation, not automation: Activities and needs of fact-checkers as a basis for designing human-centered ai systems. *Preprint*, arXiv:2211.12143.

Israa Jaradat, Pepa Gencheva, Alberto Barrón-Cedeño, Lluís Màrquez, and Preslav Nakov. 2018. ClaimRank: Detecting check-worthy claims in Arabic and English. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 26–30, New Orleans, Louisiana. Association for Computational Linguistics.

Ashkan Kazemi, Kiran Garimella, Devin Gaffney, and Scott A. Hale. 2021. Claim matching beyond English to scale global fact-checking. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4504–4517, Online. Association for Computational Linguistics.

Miaoran Li, Baolin Peng, Michel Galley, Jianfeng Gao, and Zhu Zhang. 2024. Self-checker: Plug-and-play modules for fact-checking with large language models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 163–181, Mexico City, Mexico. Association for Computational Linguistics.

Mistral AI Team. 2024. Large enough.

Anna Neumann, Dorothea Kolossa, and Robert M Nickel. 2023. Deep learning-based claim matching with multiple negatives training. In *Proceedings of the 6th International Conference on Natural Language and Speech Processing (ICNLSP 2023)*, pages 134–139, Online. Association for Computational Linguistics.

Matúš Pikuliak, Ivan Srba, Robert Moro, Timo Hromadka, Timotej Smoleň, Martin Melišek, Ivan Vykopal, Jakub Simko, Juraj Podroužek, and Maria Bielikova. 2023. Multilingual previously fact-checked claim retrieval. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16477–16500, Singapore. Association for Computational Linguistics.

Vipula Rawte, Amit Sheth, and Amitava Das. 2023. A survey of hallucination in large foundation models. *Preprint*, arXiv:2309.05922.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. 2020. MLSUM: The multilingual summarization corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8051–8067, Online. Association for Computational Linguistics.

Shaden Shaar, Nikolay Babulkov, Giovanni Da San Martino, and Preslav Nakov. 2020. That is a known lie: Detecting previously fact-checked claims. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3607–3618, Online. Association for Computational Linguistics.

Shaden Shaar, Nikola Georgiev, Firoj Alam, Giovanni Da San Martino, Aisha Mohamed, and Preslav Nakov. 2022. Assisting the human fact-checkers: Detecting all previously fact-checked claims in a document. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2069–2080, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Zien Sheikh Ali, Watheq Mansour, Fatima Haouari, Maram Hasanain, Tamer Elsayed, and Abdulaziz Al-Ali. 2023. Tahaqqaq: A real-time system for assisting twitter users in arabic claim verification. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 3170–3174, New York, NY, USA. Association for Computing Machinery.

Michael Shliselberg and Shiri Dori-Hacohen. 2022. Riet lab at checkthat!-2022: Improving decoder based re-ranking for claim matching. In *CLEF (Working Notes)*, pages 671–678.

Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. 2019. defend: Explainable fake news detection. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, page 395–405, New York, NY, USA. Association for Computing Machinery.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Ivan Vykopal, Matúš Pikuliak, Simon Ostermann, Tatiana Anikina, Michal Gregor, and Marián Šimko. 2025. Large language models for multilingual previously fact-checked claim detection. *Preprint*, arXiv:2503.02737.

Ivan Vykopal, Matúš Pikuliak, Simon Ostermann, and Marián Šimko. 2024. Generative large language models in automated fact-checking: A survey. *Preprint*, arXiv:2407.02351.

Jiaan Wang, Yunlong Liang, Fandong Meng, Beiqi Zou, Zhixu Li, Jianfeng Qu, and Jie Zhou. 2023. Zero-shot cross-lingual summarization via large language models. In *Proceedings of the 4th New Frontiers in Summarization Workshop*, pages 12–23, Singapore. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 43 others. 2024. Qwen2 technical report. *Preprint*, arXiv:2407.10671.

Arkaitz Zubiaga, Ahmet Aker, Kalina Bontcheva, Maria Liakata, and Rob Procter. 2018. Detection and resolution of rumours in social media: A survey. *ACM Comput. Surv.*, 51(2).

A Computational Resources

For our experiments, we leveraged a computational infrastructure consisting of A40 PCIe 40GB, A100 80GB and H100 NVL 94GB NVIDIA GPUs. In addition, we used API from Anthropic to run the experiments with Claude 3.5 Sonnet and Azure for deploying GPT-4o.

The experiments with TEMs took around 30 GPU hours. Our experiments with summarization and comparison of various quantization variants required approximately 600 GPU hours. Finally, the experiments with the overall pipeline – with veracity prediction – took around 400 GPU hours.

B Dataset Statistics

B.1 MultiClaim Dataset

For our experiments, we selected ***MultiClaim*** (Pikuliak et al., 2023), the most comprehensive multilingual dataset for previously fact-checked claim retrieval. We used the full dataset for retrieval experiments, as described in Section 4. For other components, we worked with a subset of ***MultiClaim***, selecting a set of 10 languages that included both high- and low-resource languages. From each language, we sampled 100 social media posts for each language while aiming to balance the distribution of veracity labels. However, the original ***MultiClaim*** dataset is highly imbalanced, with a predominant number of false social media posts. As a result, our subset contains a significant proportion of false claims. Table 6 provides detailed statistics on the subset used in our experiments.

Language	Lang. Code	Average WC	# False	# True	# Unverifiable
Czech	cs	168.60 ± 242.44	100	0	0
German	de	86.08 ± 84.90	94	1	5
English	en	111.11 ± 142.39	92	5	3
French	fr	109.14 ± 129.62	95	4	1
Hindi	hi	63.36 ± 108.82	95	4	1
Hungarian	hu	123.73 ± 178.21	97	0	3
Polish	pl	102.00 ± 130.70	96	2	2
Portuguese	pt	92.25 ± 176.08	76	6	18
Slovak	sk	126.59 ± 214.57	100	0	0
Spanish	es	95.73 ± 130.48	35	33	32
Total			880	55	65

Table 6: Statistics of a subset of ***MultiClaim*** dataset used for experiments with filtration and veracity prediction. We provide the average word count (WC) with standard deviation and the number of false, true and unverifiable claims per language.

The final veracity ratings were derived from fact-checking articles linked to particular posts. We manually evaluated these links to ensure they were correctly extracted from the metadata of the corresponding fact-checks.

B.2 AFP-Sum Dataset

To assess the ability of LLMs to summarize fact-checking articles, we collected data from the AFP organization. Specifically, we extracted fact-checking articles in 23 languages, listed in Table 7, which also includes the number of articles per language. Our dataset comprises fact-checking articles published up until September 2023.

For the final evaluation, we employed only a subset of the data, especially we used 100 fact-checking articles per language, which we randomly sampled from the ***AFP-Sum*** dataset. The statistics of the sampled dataset, consisting of 2300 fact-checking articles in 23 languages, are shown in Table 8. Besides the number of fact-checking articles, we provide the average word count for the article and for the summary along with the standard deviation.

Since the extracted summaries are in the original language, we employed Google Translate API to translate the summaries into English, which we then used for the final evaluation and calculating

Language	Lang. Code	Domain	# Articles
English	en	https://factcheck.afp.com	6358
Spanish	es	https://factual.afp.com	3999
French	fr	https://factuel.afp.com	2883
Portuguese	pt	https://checamos.afp.com	1320
German	de	https://faktencheck.afp.com	564
Indonesian	id	https://periksafakta.afp.com	506
Polish	pl	https://sprawdzam.afp.com	386
Korean	ko	https://factcheckkorea.afp.com	359
Thai	th	https://factcheckthailand.afp.com	349
Serbian	sr	https://cinjenice.afp.com	306
Finnish	fi	https://faktantarkistus.afp.com	289
Malay	ms	https://semakanfakta.afp.com	233
Slovak	sk	https://fakty.afp.com	226
Czech	cs	https://napravoumiru.afp.com	216
Dutch	nl	https://factchecknederland.afp.com	192
Bulgarian	bg	https://proveri.afp.com	139
Bengali	bn	https://factcheckbangla.afp.com	136
Romanian	ro	https://verificat.afp.com	135
Burmese	my	https://factcheckmyanmar.afp.com	128
Hindi	hi	https://factcheckhindi.afp.com	125
Greek	el	https://factcheckgreek.afp.com	121
Hungarian	hu	https://tenykerdes.afp.com	112
Catalan	ca	https://comprovem.afp.com	110

Table 7: Statistics of the ***AFP-Sum*** dataset, consisting of the languages, language codes, domains and the number of articles per language.

Lang. Code	Language	# Articles	Average WC Article	Average WC Summary
bg	Bulgarian	100	965.66 ± 533.28	81.57 ± 20.09
bn*	Bengali	100	308.93 ± 114.53	55.07 ± 17.23
ca*	Catalan	100	822.30 ± 454.67	82.69 ± 18.19
cs	Czech	100	691.35 ± 353.20	62.31 ± 14.28
de	German	100	869.32 ± 510.19	62.19 ± 15.57
el	Greek	100	1116.24 ± 500.74	86.51 ± 17.89
en	English	100	463.63 ± 197.19	58.18 ± 13.51
es	Spanish	100	713.13 ± 477.01	75.87 ± 18.69
fi*	Finnish	100	754.15 ± 369.82	57.50 ± 17.54
fr	French	100	659.96 ± 568.90	61.38 ± 23.46
hi	Hindi	100	507.20 ± 142.50	78.07 ± 17.16
hu	Hungarian	100	884.79 ± 570.36	78.02 ± 17.50
id*	Indonesian	100	458.79 ± 173.84	56.58 ± 12.63
ko	Korean	100	309.15 ± 131.40	46.99 ± 11.12
ms	Malay	100	521.20 ± 163.88	59.05 ± 13.16
my	Burmese	100	233.89 ± 77.18	31.18 ± 10.57
nl	Dutch	100	998.47 ± 515.52	73.51 ± 19.30
pl	Polish	100	836.52 ± 474.79	59.31 ± 17.34
pt	Portuguese	100	715.00 ± 343.31	80.21 ± 15.72
ro	Romanian	100	1156.78 ± 566.54	88.75 ± 19.20
sk	Slovak	100	850.55 ± 552.95	62.53 ± 22.07
sr	Serbian	100	954.83 ± 497.00	71.55 ± 19.63
th	Thai	100	121.34 ± 42.42	10.71 ± 4.68

Table 8: Statistics of the dataset used for summarization experiments, consisting of 100 fact-check articles across 23 languages. Languages marked with * are not included in other experiments besides summarization. The Arabic language is missing, which is used in other experiments.

the BERTScore and ROUGE-L.

C Retrieval Experiments

Table 9 provides the results of the experiments with simple retrieval across 20 languages, where we aimed to evaluate how accurate TEMs are for retrieving the relevant fact-checks based on the content of the social media post. We report S@10 as the main metric for the evaluation.

C.1 Criteria-based Retrieval

Figure 5 illustrates the template used for filtered retrieval experiments. Each fact-check is structured using this template, which includes the fact-checked claim, the language of the fact-checking article, the publication date, and the fact-checking organization. This structure representation is then embedded using the selected TEM.

To retrieve relevant fact-checks based on the instruction in natural language, we test different retrieval conditions, such as filtering by language or by a specific named entity. Once we obtain a list of fact-checks with a similarity score above 0.8, we perform a second retrieval step based on the content of a social media post. In this step, each fact-check is represented only by the fact-checked claim without any metadata, and is embedded using a specific TEM to facilitate retrieval.

D Summarization Experiments

Figure 6 illustrate the final prompt formats used in our summarization experiments. We present both the *Article last* and *Article first* variants.

Table 10 presents the results of the summarization experiments using various open-source and closed-source LLMs across 23 languages, evaluated with the BERTScore metric. For each LLM, we report performance in two settings: when the article is provided before the instruction (*Article first*) and when the article is provided after the instruction (*Article last*).

Similarly, Table 11 summarizes the results based on the ROUGE-L metric.

In addition to evaluating the two settings, we also examined the impact of different quantization variants on LLM performance. Specifically, we compared non-quantized models with versions quantized to 4-bit and 8-bit precision. For these experiments, we selected `Llama` models, focusing on `Llama 3.1 70B` and `Llama3.2` in 1B and 3B variants. The BERTscore results across 23 languages are presented in Table 12, while Table 13 reports ROUGE-L scores.

Model	Ver.	ara	bul	ces	deu	ell	eng	fra	hbs	hin	hun	kor	msa	mya	nld	pol	por	ron	slk	spa	tha	Avg.
BM25	Og	0.75	0.71	0.70	0.63	0.61	0.63	0.74	0.46	0.61	0.49	0.58	0.75	0.31	0.56	0.56	0.77	0.70	0.78	0.73	0.31	0.62
English TEMs																						
DistilRoBERTa	En	0.79	0.86	0.88	0.58	0.73	0.64	0.79	0.65	0.65	0.82	0.82	0.75	0.77	0.72	0.65	0.64	0.86	0.85	0.72	0.89	0.75
MiniLM-L6	En	0.84	0.89	0.85	0.64	0.80	0.69	0.82	0.70	0.75	0.87	0.84	0.78	0.79	0.76	0.70	0.70	0.86	0.84	0.77	**0.90**	0.79
MiniLM-L12	En	0.84	**0.90**	0.86	0.64	0.80	0.70	0.82	0.72	0.77	0.86	0.83	0.78	0.80	0.73	0.72	0.71	0.86	0.86	0.78	0.89	0.79
MPNet-Base	En	0.80	0.87	**0.89**	0.57	0.77	0.68	0.81	0.70	0.72	0.87	0.80	0.79	0.80	0.74	0.67	0.67	0.86	0.85	0.75	0.88	0.77
GTE-Large-En	En	0.82	0.88	0.88	0.65	0.82	0.73	0.84	0.72	0.74	0.85	0.84	0.76	0.81	**0.78**	0.71	0.69	0.87	0.86	0.79	0.89	0.80
GTR-T5-Large	En	**0.86**	0.86	0.88	**0.69**	**0.83**	**0.77**	**0.86**	**0.74**	**0.79**	**0.89**	**0.86**	**0.82**	**0.88**	**0.78**	**0.74**	**0.80**	**0.88**	**0.87**	**0.84**	**0.90**	**0.83**
Multilingual TEMs																						
BGE-M3	Og	**0.84**	0.87	0.90	0.74	0.80	0.69	**0.87**	0.67	**0.82**	0.89	0.90	0.86	**0.86**	0.74	0.72	0.79	0.88	**0.89**	0.84	**0.93**	0.82
DistilUSE-Base-Multilingual	Og	0.74	0.81	0.71	0.50	0.60	0.56	0.69	0.57	0.53	0.78	0.74	0.60	0.62	0.61	0.60	0.58	0.80	0.77	0.64	0.72	0.66
LaBSE	Og	0.77	0.84	0.81	0.48	0.70	0.44	0.72	0.57	0.56	0.82	0.77	0.67	0.77	0.61	0.57	0.66	0.78	0.74	0.64	0.79	0.69
Multilingual E5 Small	Og	0.81	0.89	0.82	0.71	0.80	0.61	0.80	0.63	0.72	0.87	0.85	0.77	0.69	0.72	0.71	0.76	0.89	0.83	0.81	0.89	0.78
Multilingual E5 Base	Og	0.81	0.87	0.85	0.70	0.77	0.64	0.83	0.60	0.67	0.88	0.86	0.80	0.74	0.73	0.66	0.77	0.88	0.84	0.81	0.89	0.78
Multilingual E5 Large	Og	**0.84**	**0.90**	**0.92**	**0.78**	**0.82**	**0.75**	0.86	**0.74**	0.81	**0.90**	**0.91**	**0.88**	0.81	**0.83**	**0.77**	**0.82**	**0.90**	**0.89**	**0.87**	0.85	**0.84**
MiniLM-L12-Multilingual	Og	0.49	0.83	0.75	0.48	0.58	0.58	0.66	0.55	0.49	0.79	0.61	0.54	0.58	0.64	0.61	0.51	0.79	0.77	0.57	0.81	0.63
MPNet-Base-Multilingual	Og	0.70	0.81	0.78	0.53	0.63	0.61	0.73	0.56	0.63	0.83	0.71	0.62	0.75	0.66	0.60	0.57	0.84	0.80	0.64	0.86	0.69

Table 9: TEM results for retrieving previously fact-checked claims across 20 languages using the S@10 metric. The best scores for each configuration – English translation (En) or original language (Og) – are in bold. GTR-T-Large performed best on English translations, while Multilingual E5 Large excelled on multilingual data, surpassing English TEMs.

Model	Version	Quant.	bg	bn	ca	cs	de	el	en	es	fi	fr	hi	hu	id	ko	ms	my	nl	pl	pt	ro	sk	sr	th	Avg.
Open-Source LLMs																										
C4AI Command R+	Article first	4bit	**0.75**	0.76	**0.74**	0.74	**0.75**	**0.76**	0.76	0.73	**0.75**	**0.74**	0.76	**0.75**	**0.77**	**0.77**	**0.76**	**0.74**	**0.75**	**0.75**	**0.74**	**0.76**	**0.75**	**0.75**	**0.75**	**0.75**
	Article last	4bit	0.70	0.71	0.72	0.69	0.71	0.67	0.76	0.70	0.71	0.69	0.71	0.68	0.72	0.75	0.74	0.66	0.73	0.70	0.72	0.69	0.73	0.73	0.64	0.71
Llama3.1 70B Instruct	Article first	4bit	**0.75**	**0.77**	**0.74**	**0.74**	**0.75**	**0.76**	**0.76**	0.73	**0.75**	**0.74**	0.76	**0.75**	**0.77**	**0.77**	**0.76**	0.72	**0.75**	**0.75**	**0.74**	**0.76**	**0.75**	**0.75**	**0.75**	**0.75**
	Article last	4bit	**0.75**	**0.77**	**0.74**	**0.74**	**0.75**	**0.76**	**0.76**	**0.74**	0.74	**0.74**	**0.77**	0.69	**0.77**	**0.77**	**0.76**	0.72	**0.75**	**0.75**	**0.74**	0.75	**0.75**	**0.75**	**0.75**	**0.75**
Llama3.3 70B Instruct	Article first	4bit	0.74	0.76	0.73	**0.74**	0.74	0.75	0.75	0.73	**0.75**	0.73	0.76	**0.75**	0.76	0.76	0.75	0.70	**0.75**	0.74	**0.74**	**0.76**	0.74	0.74	**0.75**	0.74
	Article last	4bit	**0.75**	0.76	0.73	**0.74**	0.74	0.75	0.75	0.73	**0.75**	0.73	0.76	0.74	0.76	0.76	0.75	0.70	**0.75**	0.74	0.73	0.75	0.74	0.74	**0.75**	0.74
Mistral Large	Article first	4bit	**0.75**	**0.77**	0.73	**0.74**	**0.75**	**0.76**	**0.76**	0.73	**0.75**	**0.74**	0.75	**0.75**	**0.77**	**0.77**	**0.76**	**0.74**	**0.75**	**0.75**	**0.74**	0.75	0.74	0.74	**0.75**	**0.75**
	Article last	4bit	**0.75**	0.76	**0.74**	**0.74**	**0.75**	0.75	**0.76**	0.73	**0.75**	**0.74**	0.75	0.74	**0.77**	**0.77**	**0.76**	**0.74**	**0.75**	0.74	**0.74**	0.75	0.74	0.74	**0.75**	**0.75**
Qwen2 72B Instruct	Article first	4bit	0.74	0.75	0.73	0.73	0.74	0.75	0.75	0.72	0.74	0.73	0.75	0.73	0.76	0.76	0.74	**0.74**	0.74	0.74	0.73	0.74	0.74	0.74	0.74	0.74
	Article last	4bit	0.74	0.76	0.73	0.73	0.74	0.74	0.75	0.72	0.74	0.73	0.75	0.74	0.76	0.76	0.74	0.73	0.74	0.74	0.73	0.75	0.74	0.73	0.74	0.74
Qwen2.5 0.5B Instruct	Article first	-	0.70	0.68	0.71	0.70	0.72	0.69	0.74	0.71	0.68	0.71	0.69	0.69	0.73	0.72	0.71	0.68	0.71	0.71	0.71	0.71	0.70	0.69	0.72	0.70
	Article last	-	0.70	0.68	0.70	0.70	0.71	0.68	0.73	0.70	0.68	0.70	0.68	0.69	0.73	0.72	0.71	0.67	0.70	0.70	0.70	0.71	0.70	0.69	0.72	0.70
Qwen2.5 1.5B Instruct	Article first	-	0.73	0.73	0.73	0.72	0.73	0.72	0.75	0.73	0.71	0.73	0.73	0.72	0.76	0.74	0.74	0.70	0.74	0.73	0.73	0.74	0.73	0.72	0.74	0.73
	Article last	-	0.73	0.73	0.72	0.72	0.73	0.72	0.75	0.72	0.72	0.72	0.73	0.72	0.75	0.75	0.74	0.69	0.73	0.73	0.72	0.73	0.72	0.72	0.74	0.73
Qwen2.5 3B Instruct	Article first	-	0.74	0.75	0.73	0.73	0.74	0.74	**0.76**	0.73	0.73	0.73	0.75	0.73	0.76	0.75	0.75	0.71	0.74	0.74	**0.74**	0.75	0.74	0.74	0.74	0.74
	Article last	-	0.71	0.71	0.71	0.69	0.71	0.71	0.74	0.71	0.71	0.72	0.72	0.71	0.72	0.74	0.72	0.68	0.71	0.69	0.71	0.72	0.69	0.68	0.73	0.71
Qwen2.5 7B Instruct	Article first	-	0.73	0.75	0.73	0.73	0.74	0.74	0.75	0.72	0.73	0.73	0.74	0.73	0.76	0.75	0.75	0.70	0.73	0.73	0.73	0.74	0.73	0.73	0.74	0.74
	Article last	-	0.74	0.75	0.73	0.73	0.74	0.74	**0.76**	0.73	0.74	0.73	0.74	0.73	0.76	0.75	0.75	0.72	0.74	0.74	**0.74**	0.75	0.74	0.74	**0.75**	0.74
Qwen2.5 72B Instruct	Article first	4bit	**0.75**	**0.77**	**0.74**	**0.74**	0.74	0.75	**0.76**	0.73	**0.75**	**0.74**	0.76	0.74	**0.77**	0.76	**0.76**	**0.74**	**0.75**	**0.75**	**0.74**	0.75	**0.75**	**0.75**	**0.75**	**0.75**
	Article last	4bit	**0.75**	0.76	**0.74**	**0.74**	0.74	0.75	0.75	0.73	**0.75**	**0.74**	0.76	**0.75**	**0.77**	0.76	**0.76**	**0.74**	**0.75**	0.74	**0.74**	**0.76**	0.74	**0.75**	**0.75**	**0.75**
Gemma3 27B	Article first	4bit	0.72	0.75	0.72	0.72	0.73	0.74	0.74	0.72	0.73	0.73	0.74	0.73	0.75	0.75	0.74	0.73	0.73	0.72	0.72	0.73	0.73	0.72	0.73	0.73
	Article last	4bit	0.72	0.74	0.72	0.73	0.73	0.74	0.74	0.72	0.73	0.72	0.74	0.73	0.75	0.74	0.74	0.72	0.73	0.73	0.72	0.73	0.73	0.73	0.73	0.73
Closed-Source LLMs																										
Claude 3.5 Sonnet	Article first	-	0.74	0.76	0.73	**0.74**	0.73	0.74	0.75	0.72	0.74	0.73	0.75	0.74	0.76	0.75	0.75	0.73	0.74	0.74	0.73	0.74	0.74	0.73	**0.75**	0.74
	Article last	-	0.74	0.76	0.73	**0.74**	0.74	0.75	**0.76**	0.73	0.74	**0.74**	0.75	0.74	0.76	0.76	0.75	**0.74**	0.74	0.74	**0.74**	0.74	0.74	0.74	**0.75**	0.74
GPT-4o	Article first	-	0.74	0.76	0.73	**0.74**	0.74	0.75	0.75	0.72	0.74	0.73	0.75	0.74	0.76	0.76	0.75	**0.74**	0.74	0.74	**0.74**	0.74	0.74	0.74	**0.75**	0.74
	Article last	-	0.74	0.76	0.73	0.73	0.74	0.75	0.75	0.72	0.74	0.73	0.75	0.74	0.76	0.76	0.75	**0.74**	0.74	0.74	0.73	0.74	0.74	0.74	**0.75**	0.74

Table 10: BERTScore evaluation of summarization performance across 23 languages for various LLMs in two settings: *Article first* and *Article last*. The best results for each language are in bold.

Model	Version	Quant.	bg	bn	ca	cs	de	el	en	es	fi	fr	hi	hu	id	ko	ms	my	nl	pl	pt	ro	sk	sr	th	Avg.
Open-Source LLMs																										
C4AI Command R+	Article first	4bit	**0.32**	0.34	0.28	0.28	0.30	**0.32**	**0.33**	0.27	0.30	**0.28**	0.34	0.30	**0.37**	0.34	**0.35**	**0.28**	0.31	**0.30**	0.30	0.32	0.29	**0.30**	0.32	**0.31**
	Article last	4bit	0.12	0.19	0.18	0.12	0.12	0.08	0.32	0.08	0.19	0.05	0.16	0.10	0.11	0.28	0.24	0.15	0.23	0.14	0.11	0.06	0.25	0.26	0.09	0.16
Llama3.1 70B Instruct	Article first	4bit	0.31	0.34	**0.29**	**0.29**	**0.31**	**0.32**	**0.33**	**0.29**	**0.31**	**0.28**	0.34	**0.31**	**0.37**	0.34	0.34	0.24	**0.32**	**0.30**	**0.31**	**0.33**	0.28	**0.30**	0.33	**0.31**
	Article last	4bit	0.30	0.32	0.28	0.28	**0.31**	0.30	**0.33**	0.28	0.28	0.27	0.33	0.13	0.34	0.34	0.32	0.24	0.31	**0.30**	0.28	0.27	0.28	0.29	0.32	0.29
Llama3.3 70B Instruct	Article first	4bit	0.30	**0.35**	**0.29**	**0.29**	**0.31**	**0.32**	0.32	0.28	**0.31**	0.27	**0.35**	0.30	**0.37**	0.34	0.34	0.22	0.31	**0.30**	0.30	**0.33**	0.29	0.29	**0.34**	**0.31**
	Article last	4bit	0.31	0.34	**0.29**	0.28	**0.31**	0.31	0.32	**0.29**	**0.31**	0.27	**0.35**	0.26	0.35	**0.35**	0.34	0.22	0.31	0.29	0.30	0.31	0.29	0.29	**0.34**	**0.31**
Mistral Large	Article first	4bit	0.31	0.33	0.27	0.28	0.30	**0.32**	**0.33**	0.27	0.30	0.27	0.32	0.30	0.35	0.34	0.33	0.27	0.31	0.29	0.30	0.30	0.28	0.29	0.32	0.30
	Article last	4bit	0.30	0.33	0.28	**0.29**	**0.31**	**0.32**	0.31	0.27	**0.31**	0.27	0.33	0.29	0.36	0.34	0.33	**0.28**	0.31	0.29	0.29	0.31	**0.30**	0.29	0.33	**0.31**
Qwen2 72B Instruct	Article first	4bit	0.28	0.31	0.25	0.26	0.27	0.29	0.29	0.25	0.28	0.24	0.30	0.27	0.32	0.31	0.30	0.26	0.28	0.27	0.26	0.28	0.27	0.27	0.29	0.28
	Article last	4bit	0.28	0.32	0.25	0.26	0.28	0.29	0.30	0.24	0.28	0.24	0.29	0.28	0.32	0.31	0.29	0.26	0.28	0.27	0.27	0.28	0.26	0.27	0.30	0.28
Qwen2.5 0.5B Instruct	Article first	-	0.19	0.16	0.19	0.18	0.24	0.16	0.25	0.21	0.15	0.19	0.17	0.16	0.25	0.23	0.21	0.15	0.21	0.19	0.20	0.19	0.17	0.16	0.24	0.19
	Article last	-	0.20	0.16	0.18	0.19	0.21	0.16	0.25	0.18	0.15	0.19	0.18	0.16	0.24	0.24	0.22	0.15	0.18	0.19	0.17	0.19	0.17	0.17	0.24	0.19
Qwen2.5 1.5B Instruct	Article first	-	0.25	0.24	0.24	0.22	0.25	0.23	0.26	0.25	0.21	0.24	0.26	0.21	0.31	0.26	0.28	0.17	0.27	0.25	0.25	0.27	0.23	0.23	0.29	0.25
	Article last	-	0.26	0.24	0.22	0.21	0.24	0.23	0.26	0.25	0.23	0.23	0.25	0.21	0.29	0.28	0.27	0.17	0.25	0.24	0.23	0.26	0.24	0.23	0.29	0.24
Qwen2.5 3B Instruct	Article first	-	0.27	0.28	0.25	0.24	0.27	0.27	0.30	0.25	0.26	0.24	0.28	0.25	0.31	0.30	0.28	0.21	0.27	0.26	0.26	0.28	0.25	0.26	0.30	0.27
	Article last	-	0.24	0.25	0.23	0.19	0.25	0.23	0.28	0.23	0.21	0.23	0.24	0.22	0.27	0.29	0.26	0.16	0.25	0.20	0.23	0.25	0.18	0.17	0.29	0.23
Qwen2.5 7B Instruct	Article first	-	0.25	0.28	0.23	0.24	0.26	0.26	0.26	0.23	0.24	0.23	0.27	0.25	0.30	0.26	0.27	0.17	0.25	0.25	0.24	0.26	0.25	0.24	0.27	0.25
	Article last	-	0.28	0.30	0.25	0.26	0.28	0.28	0.29	0.25	0.27	0.24	0.29	0.25	0.32	0.31	0.30	0.23	0.28	0.26	0.27	0.29	0.27	0.27	0.31	0.28
Qwen2.5 72B Instruct	Article first	4bit	0.29	0.32	0.27	0.28	0.29	0.29	0.30	0.26	0.30	0.25	0.31	0.29	0.34	0.32	0.32	0.26	0.29	0.29	0.28	0.29	0.28	0.28	0.31	0.29
	Article last	4bit	0.29	0.32	0.26	0.27	0.28	0.30	0.30	0.25	0.30	0.25	0.31	0.29	0.34	0.32	0.31	0.25	0.29	0.26	0.28	0.29	0.28	0.28	0.29	0.31	0.29
Gemma3 27B	Article first	4bit	0.26	0.30	0.24	0.25	0.26	0.26	0.29	0.24	0.26	0.25	0.30	0.26	0.32	0.29	0.31	0.26	0.27	0.25	0.25	0.26	0.25	0.25	0.30	0.27
	Article last	4bit	0.26	0.30	0.24	0.25	0.27	0.27	0.28	0.24	0.26	0.24	0.30	0.27	0.31	0.29	0.30	0.25	0.26	0.25	0.25	0.27	0.25	0.26	0.30	0.27
Closed-Source LLMs																										
Claude 3.5 Sonnet	Article first	-	0.30	0.34	0.26	0.28	0.29	0.29	0.30	0.26	0.29	0.27	0.33	0.30	0.33	0.32	0.31	0.27	0.29	0.28	0.28	0.30	0.28	0.27	0.33	0.29
	Article last	-	0.29	0.33	0.27	**0.29**	0.30	0.30	0.32	0.27	0.29	0.27	0.33	0.28	0.34	0.34	0.31	**0.28**	0.29	0.29	0.29	0.29	0.28	0.28	0.33	0.30
GPT 4o	Article first	-	0.29	0.32	0.27	0.28	0.29	0.30	0.30	0.26	0.29	0.27	0.33	0.28	0.33	0.33	0.31	0.27	0.28	0.28	0.28	0.29	0.28	0.27	0.31	0.29
	Article last	-	0.29	0.33	0.26	0.27	0.29	0.29	0.30	0.26	0.29	0.25	0.32	0.29	0.32	0.33	0.31	0.26	0.28	0.27	0.27	0.29	0.27	0.28	0.31	0.29

Table 11: ROUGE-L evaluation of summarization performance across 23 languages for various LLMs in two settings: *Article first* and *Article last*. The best results for each language are in bold.

Model	Version	Quant.	bg	bn	ca	cs	de	el	en	es	fi	fr	hi	hu	id	ko	ms	my	nl	pl	pt	ro	sk	sr	th	Avg.
Llama3.2 1B Instruct	Article first	4bit	0.70	0.67	0.71	0.69	0.71	0.70	0.74	0.70	0.66	0.69	0.64	0.64	0.72	0.72	0.70	0.65	0.72	0.68	0.70	0.70	0.69	0.68	0.60	0.69
		8bit	**0.72**	**0.72**	0.71	**0.71**	**0.73**	**0.73**	**0.74**	**0.71**	**0.70**	**0.72**	**0.72**	**0.67**	**0.74**	**0.73**	**0.73**	0.66	**0.73**	**0.71**	**0.72**	0.72	**0.72**	**0.71**	**0.72**	**0.72**
		-	**0.72**	**0.72**	**0.72**	**0.71**	**0.73**	**0.73**	**0.74**	**0.71**	**0.70**	0.71	**0.72**	0.66	**0.74**	**0.73**	**0.73**	**0.67**	0.72	**0.71**	0.71	**0.73**	0.71	**0.71**	**0.72**	**0.72**
	Article last	4bit	0.61	0.60	0.66	0.64	0.67	0.62	0.74	0.68	0.61	0.67	0.63	0.63	0.69	0.63	0.67	0.52	0.67	0.65	0.67	0.65	0.61	0.62	0.59	0.64
		8bit	0.64	0.63	0.68	0.67	0.70	0.64	0.75	0.69	0.64	0.70	0.64	0.65	0.71	0.69	0.69	0.53	0.69	0.67	0.69	0.67	0.65	0.65	0.61	0.66
		-	0.64	0.63	0.69	0.67	0.70	0.64	0.75	0.70	0.64	0.70	0.64	0.64	0.71	0.69	0.69	0.53	0.69	0.67	0.69	0.67	0.65	0.66	0.61	0.66
Llama3.2 3B Instruct	Article first	4bit	**0.74**	0.75	**0.73**	**0.73**	**0.74**	0.74	**0.76**	**0.73**	0.72	**0.73**	**0.75**	0.69	**0.76**	0.75	0.74	0.69	**0.74**	**0.74**	**0.72**	**0.75**	**0.73**	**0.73**	**0.74**	0.73
		8bit	**0.74**	0.75	**0.73**	**0.73**	**0.74**	0.74	0.75	0.72	**0.74**	**0.73**	**0.75**	**0.70**	0.75	0.75	0.74	**0.70**	**0.74**	0.73	**0.72**	**0.75**	**0.73**	**0.73**	**0.74**	**0.74**
		-	0.73	0.75	**0.73**	**0.73**	**0.74**	**0.75**	**0.76**	0.72	**0.74**	**0.73**	**0.75**	0.69	0.75	**0.76**	0.74	**0.70**	**0.74**	**0.74**	**0.72**	**0.75**	**0.73**	**0.73**	**0.74**	**0.74**
	Article last	4bit	0.73	0.75	0.72	0.72	0.73	0.73	0.76	0.72	0.70	**0.73**	**0.75**	0.66	**0.76**	**0.76**	**0.75**	0.68	0.73	0.72	0.71	0.73	0.72	0.72	**0.74**	0.73
		8bit	0.70	**0.76**	0.71	0.69	0.72	0.66	0.75	0.72	0.67	**0.73**	0.73	0.66	0.74	**0.76**	0.73	0.69	0.72	0.71	0.71	0.69	0.68	0.69	0.73	0.71
		-	0.69	**0.76**	0.70	0.69	0.72	0.67	0.75	0.72	0.66	**0.73**	0.73	0.66	0.74	**0.76**	0.73	0.70	0.72	0.71	0.71	0.69	0.68	0.69	**0.74**	0.71
Llama3.1 70B Instruct	Article first	4bit	**0.75**	**0.77**	**0.74**	**0.74**	**0.75**	**0.76**	**0.76**	**0.73**	**0.75**	**0.74**	**0.76**	**0.75**	**0.77**	**0.77**	**0.76**	0.72	**0.75**	**0.75**	**0.74**	**0.76**	**0.75**	**0.75**	**0.75**	**0.75**
		-	**0.75**	**0.77**	**0.74**	**0.74**	0.74	0.75	**0.76**	**0.73**	**0.75**	**0.74**	**0.76**	**0.75**	**0.77**	**0.77**	**0.76**	**0.75**	**0.75**	**0.75**	**0.74**	0.75	0.74	**0.75**	**0.75**	**0.75**

Table 12: BERTScore evaluation of LLM summarization across 23 languages, comparing non-quantized models with 4-bit and 8-bit quantized variants. The best results for each language are in bold.

Model	Verstion	Quant	bg	bn	ca	cs	de	el	en	es	fi	fr	hi	hu	id	ko	ms	my	nl	pl	pt	ro	sk	sr	th	All
Llama3.2 1B Instruct	Article first	4bit	0.21	0.16	0.20	0.17	0.22	0.19	0.28	0.20	0.11	0.17	0.06	0.06	0.21	0.24	0.15	0.14	0.22	0.14	0.15	0.17	0.18	0.14	0.04	0.17
		8bit	**0.24**	0.26	**0.22**	**0.23**	**0.26**	0.25	**0.29**	**0.24**	**0.19**	**0.23**	**0.26**	**0.10**	**0.29**	**0.27**	**0.27**	**0.17**	**0.26**	**0.22**	**0.24**	**0.25**	**0.22**	**0.21**	0.27	**0.24**
		-	**0.24**	**0.27**	**0.22**	0.22	**0.26**	**0.26**	**0.29**	**0.24**	**0.19**	0.22	**0.26**	0.09	**0.29**	**0.27**	**0.27**	**0.17**	**0.26**	**0.22**	**0.24**	**0.25**	**0.22**	0.20	**0.28**	0.23
	Article last	4bit	0.01	0.00	0.04	0.05	0.04	0.02	0.27	0.06	0.02	0.05	0.01	0.03	0.06	0.05	0.04	0.02	0.07	0.02	0.05	0.04	0.03	0.01	0.01	0.04
		8bit	0.03	0.02	0.08	0.08	0.11	0.03	0.26	0.11	0.03	0.12	0.03	0.04	0.14	0.16	0.09	0.05	0.09	0.07	0.09	0.05	0.06	0.04	0.04	0.08
		-	0.02	0.01	0.08	0.08	0.11	0.03	0.26	0.11	0.03	0.13	0.03	0.04	0.14	0.16	0.09	0.04	0.09	0.07	0.09	0.05	0.06	0.05	0.04	0.08
Llama3.2 3B Instruct	Article first	4bit	0.27	0.31	0.25	**0.27**	**0.28**	0.27	**0.31**	0.25	0.25	0.24	**0.32**	0.14	**0.32**	0.30	**0.30**	0.19	0.28	**0.28**	**0.22**	**0.30**	**0.26**	0.26	0.30	**0.27**
		8bit	**0.29**	0.31	0.26	**0.27**	**0.28**	**0.29**	**0.31**	**0.26**	**0.28**	**0.25**	**0.32**	**0.17**	**0.32**	0.31	**0.30**	0.21	0.28	**0.28**	0.20	**0.30**	**0.26**	**0.27**	**0.31**	**0.27**
		-	**0.29**	**0.32**	**0.27**	**0.27**	**0.28**	**0.29**	**0.31**	**0.26**	**0.28**	**0.25**	0.31	0.14	**0.32**	**0.32**	**0.30**	**0.22**	**0.29**	0.27	0.17	**0.30**	**0.26**	0.26	**0.31**	**0.27**
	Article last	4bit	0.26	0.30	0.19	0.22	0.24	0.24	0.29	0.23	0.18	0.24	0.29	0.06	0.30	0.30	0.28	0.19	0.25	0.24	0.12	0.21	0.22	0.21	0.29	0.23
		8bit	0.13	0.31	0.09	0.13	0.20	0.06	0.30	0.18	0.08	0.22	0.23	0.04	0.22	0.30	0.19	0.19	0.16	0.17	0.10	0.07	0.09	0.12	0.29	0.17
		-	0.11	0.31	0.09	0.12	0.21	0.06	0.29	0.19	0.06	0.22	0.24	0.04	0.19	0.31	0.17	0.21	0.17	0.16	0.11	0.07	0.10	0.10	0.29	0.17
Llama3.1 70B Instruct	Article first	4bit	0.31	0.34	**0.29**	0.29	**0.31**	**0.32**	**0.33**	**0.29**	0.31	**0.28**	0.34	**0.31**	**0.37**	0.34	**0.34**	0.24	**0.32**	**0.30**	**0.31**	**0.33**	0.28	**0.30**	0.33	0.31
		-	**0.32**	**0.35**	**0.29**	**0.30**	**0.31**	**0.32**	**0.33**	**0.29**	**0.32**	**0.28**	**0.36**	**0.31**	**0.37**	**0.35**	**0.34**	**0.30**	**0.32**	**0.30**	**0.31**	**0.33**	**0.29**	**0.30**	**0.34**	**0.32**

Table 13: ROUGE-L evaluation of LLM summarization across 23 languages, comparing non-quantized models with 4-bit and 8-bit quantized variants. The best results for each language are in bold.

Template

Fact-checked claim: `{claim}`
Language: `{language}` (`{language_code}`)
Published date: `{yyyy/mm/dd}`
Fact-checking organization: `{organization_name}`

An Example

Fact-checked claim: Vaccines cause autism
Language: English (en)
Published date: 2019/03/26
Fact-checking organization: healthfeedback.org

Figure 5: Template used to structure fact-checks for filtered retrieval, along with an example illustrating its format, including the fact-checked claim, language, publication date and fact-checking organization.

Article Last

Create a 3-5 sentence summary of the following article, focusing on the main idea. Provide only the summary in English without any additional text.
Article: `{text}`
Summary:

Article First

Article: `{text}`

Create a 3-5 sentence summary of the article, focusing on the main idea. Provide only the summary in English without any additional text.
Summary:

Figure 6: Prompts used for the experiments with summarization.

E Veracity Explanations

Figure 7 provides the prompt templates for each step within our pipeline. These prompts are used in the pipeline to obtain the final veracity prediction.

E.1 Error Analysis

In this section, we investigate the errors and incorrect explanations in veracity prediction. We conducted both manual inspection of a subset of incorrect predictions and automatic analysis to evaluate these errors.

Manual Analysis. For our manual investigation, we randomly selected 20 samples per LLM[9] with incorrect predicted labels, resulting in a total of 140 samples. One of the authors analyzed the retrieved relevant fact-checks and LLM-generated explanations, categorizing them into several types.

The most prevalent error category was **missing context** within claims, which occurred in 27% of our manually checked samples. This missing context made it difficult to identify relevant fact-checks and predict veracity correctly. Notably, in 63% of these cases, the LLMs provided correct explanations acknowledging the missing context.

The second most common error (16% of cases) stemmed from failures in the previous steps, where **relevant fact-checks were not identified**. In these instances, the LLMs correctly explained that none of the retrieved fact-checked information was directly relevant to the given claim, yet still produced incorrect veracity assessments.

Misunderstanding of claims and provided relevant fact-checks accounted for 17% of errors. In these cases, LLMs focused on different aspects of the provided fact-checks or failed to grasp the main point of the claim. We observed some instances where LLMs incorrectly relied on information from the social media post itself to explain its veracity, particularly with longer posts.

Another error pattern (12% of cases) involved **LLMs predicting veracity based on ratings mentioned in their generated summaries**, while the actual fact-check ratings differed. For example, a summary might characterize a claim as a hoax, while the rating extracted from the fact-check was "unverifiable".

In 4% of cases, LLMs relied only on **the rating from the first fact-check**, despite the presence of fact-checks with correct ratings later in the prompt context. This suggests an incorrect assumption that the first fact-check should be used in veracity prediction.

Finally, 15% of errors could be attributed to **ground truth issues**, primarily in cases where fact-checks classified claims as having "no evidence" – which our normalization process converted to "unverifiable". However, in all these cases, the LLMs' explanations of the claims and their veracity were correct and supported by information from the fact-check summaries.

[9] The `Gemma3` model was not included in the error analysis, as it was added into the study during the final stages, after the manual investigation had already been conducted.

Article Summary

```
Article: {document}

Create a 3-5 sentence summary of the article, focusing on the main idea. Provide only the summary in English without any additional text.
Summary:
```

Overall Summary

```
Retrieved claim: {claim1}
Summary: {article_summary1}

Retrieved claim: {claim2}
Summary: {article_summary2}

...

Generate a brief, one-paragraph summary that captures the key information from all the relevant claims and fact-checks. Ensure the summary covers the main points of each claim and addresses all the topics presented, while remaining concise and comprehensive.
```

Filtration

```
Input claim: {post_text}
----------
Claim ID: 1
Fact-checked claim: {claim1}

...

Claim ID: 50
Fact-checked claim: {claim50}
----------
Identify only fact-checked claims that are implied by the input claim. For each claim, provide the claim ID, the fact-checked claim, and an explanation of fact-checked claim's implication to the input claim.

Output Format (JSON):
{
    "fact_checked_claims": [
    {
        "claim_id": "<ClaimID1>",
        "fact_checked_claim": "<Claim1>",
        "explanation": "<Explanation of Claim1>"
    },
    {
        "claim_id": "<ClaimID2>",
        "fact_checked_claim": "<Claim2>",
        "explanation": "<Explanation of Claim2>"
    }
    ]
}
```

Veracity prediction

```
Input Claim: {post_text}
---
Fact-checked claim: {claim1}
Summary of the fact-check article: {article_summary1}

Fact-checked claim: {claim2}
Summary of the fact-check article: {article_summary2}

...
---
Based only on the provided fact-checked information that is directly relevant to the input claim, determine the veracity of the claim.
Ignore fact-checks that do not apply. The veracity should be classified as one of the following:
    - "True" if the claim is accurate based on the relevant fact-checked information.
    - "False" if the claim is inaccurate based on the relevant fact-checked information.
    - "Unverifiable" if there is insufficient or no relevant fact-checked information to assess the claim.
Provide a concise explanation that justifies your prediction.
---
Output Format (JSON):
{
    "veracity": "<True/False/Unverifiable>",
    "explanation": "<Explanation for the prediction>"
}
```

Figure 7: Prompt templates used in the pipeline for the veracity prediction.

Model	Missing FC [%]
Mistral Large	25.5
C4AI Command R+	25.3
Qwen2.5 72B	39.1
Llama3.1 70B	33.5
Llama3.3 70B	29.2
Llama3.1 8B	29.9
Qwen2.5 7B	48.6

Table 14: Percentage of posts for which no ground truth relevant fact-checks were present in the retrieved context for each LLM.

Automatic Analysis. Since one of the observed errors stems from the failure in previous steps to identify relevant fact-checks, we conducted an automatic analysis focusing on the proportion of cases where relevant fact-checks were missing from the retrieved context. Table 14 presents the percentage of posts for each model where none of the ground truth-relevant fact-checks were included in the list of relevant fact-checks. Without access to relevant fact-checks, models can struggle to accurately predict veracity regardless of their reasoning capabilities. The analysis reseals variations across LLMs, with smaller models generally exhibiting higher rates of missing fact-checks. Notably, `Qwen2.5 7B` showed the highest proportion (48.6%) of posts without relevant fact-checks, while `C4AI Command R+` and `Mistral Large` performed best with approximately 25% of posts lacking relevant fact-checks. These findings suggest that retrieval quality remains a bottleneck in the fact-checking pipeline, particularly for smaller models.

F Developed Application

The web-based application integrates the pipeline introduced in Section 3. For retrieval, we use the best-performing TEM model, `Multilingual E5`. The backend runs `Llama3.3 70B`, selected for its strong summarization capabilities and effective filtration of irrelevant fact-checks.

Our fact-check database aggregates fact-checked claims from multiple fact-checking organizations in over 80 languages. We store fact-checked claims, metadata (e.g., language, fact-checking article, rating) and calculated `Multilingual E5` embeddings of fact-checked claims in Milvus[10] vector database.

Users submit queries, and the system returns a ranked list of relevant fact-checks identified by the LLM, along with their summaries and explanations. Additionally, the system provides an overall summary, a veracity label distribution graph and an explanation of the verdict. This information supports users in making the final decision.

F.1 Interface

The developed application consists of four main components. (1) *Text input* (see Figure 8), where the user provides the claim for which the tool should return relevant fact-checking articles. (2) *List of relevant fact-checks* (see Figure 9), where we provide all the relevant fact-checks identified by the LLM. (3) *List of non-relevant fact-checks* (see Figure 10), where we list the fact-checks that were retrieved in the retrieval step but were not classified by the LLM as relevant. Since LLMs are not 100% accurate in identifying relevant fact-checks, we also provide the rest of the fact-checks to make the application robust and provide all the information that was obtained within our pipeline for fact-checkers to make the informed decision. (4) *System response* (see Figure 11), which includes the overall summary of the input claim and all relevant fact-checks, a veracity distribution graph based on the ratings of the relevant fact-checking articles and an explanation of the predicted veracity label.

[10] https://github.com/milvus-io/milvus

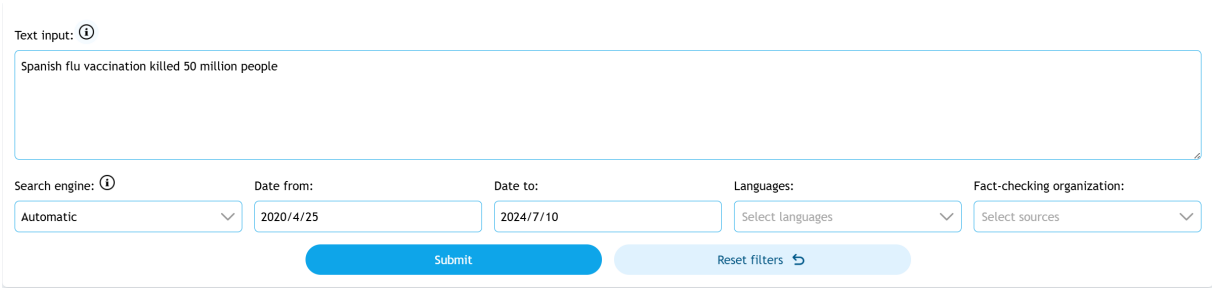

Figure 8: User interface component for the **text input**.

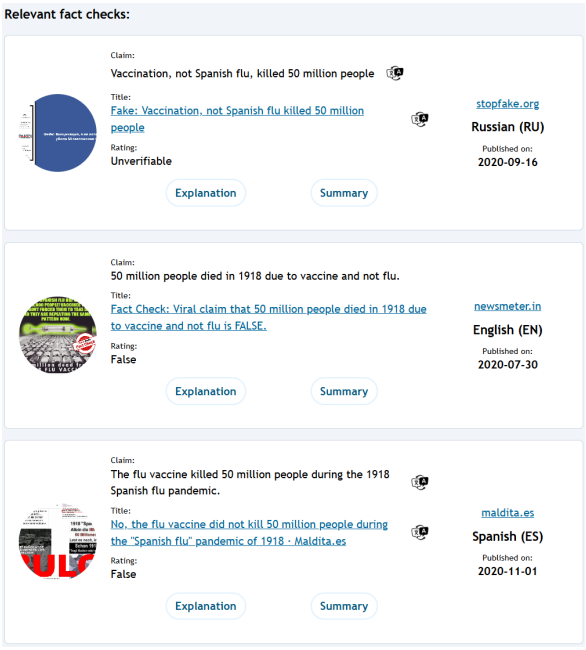

Figure 9: User interface component for a **list of relevant fact-checks** identified by the LLM within our pipeline. For each relevant fact-check, we provide the summary of the fact-checking article and an explanation of why the fact-check was classified as relevant.

Figure 10: User interface component for a **list of non-relevant fact-checks**.

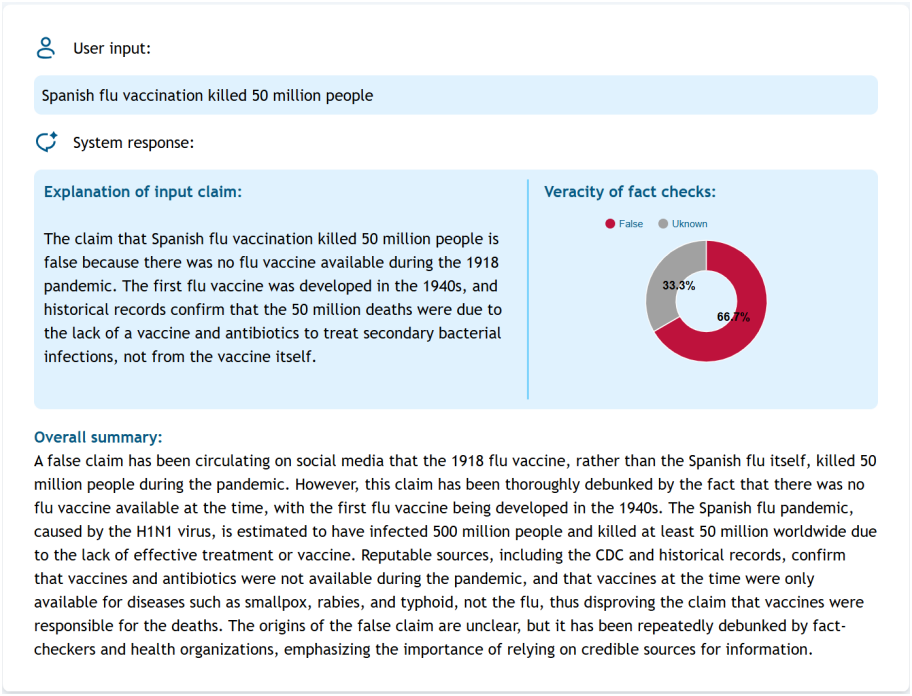

Figure 11: User interface component for **system response**, where we provide the overall summary of the claim and relevant fact-checks, a veracity distribution graph and the explanation of the predicted veracity prediction.