

dfkinit2b at CheckThat! 2025: Leveraging LLMs and Ensemble of Methods for Multilingual Claim Normalization

Notebook for the CheckThat! Lab at CLEF 2025

Tatiana Anikina^{1,†}, Ivan Vykopal^{2,3,†}, Sebastian Kula³, Ravi Kiran Chikkala⁴, Natalia Skachkova¹, Jing Yang⁵, Veronika Solopova⁵, Vera Schmitt^{1,5} and Simon Ostermann^{1,6}

¹*German Research Center for Artificial Intelligence, Saarland Informatics Campus, Germany*

²*Faculty of Information Technology, Brno University of Technology, Brno, Czech Republic*

³*Kempelen Institute of Intelligent Technologies, Bratislava, Slovakia*

⁴*Saarland University, Germany*

⁵*Technische Universität Berlin, Germany*

⁶*Centre for European Research in Trusted AI*

Abstract

The rapid spread of misinformation on social media across languages presents a major challenge for fact-checking efforts. Social media posts are often noisy, informal, and unstructured, with irrelevant content, making it difficult to extract concise, verifiable claims. To address this, the CLEF 2025 CheckThat! Shared Task on Multilingual Claim Extraction and Normalization focuses on transforming social media posts into normalized claims, short, clear and check-worthy statements that capture the essence of potentially misleading content. In this paper, we investigate several approaches to this task, including parameter-efficient fine-tuning, prompting large language models (LLMs), and an ensemble of methods. We evaluate our approaches in two settings: *monolingual*, where we are provided with training and validation data, and the *zero-shot setting*, where no training data is available for the target language. Our approaches achieved first place in 6 out of 13 languages in the *monolingual setting* and ranked second or third in the remaining languages. In the *zero-shot setting*, we achieved the highest performance across all seven languages, demonstrating strong generalization to unseen languages.

Keywords

Fact-Checking, Claim Normalization, Claim Extraction, Multilingual NLP

1. Introduction

The proliferation of false and misleading information online has emerged as a pressing global concern. Social media platforms, due to their rapid dissemination and high popularity, have become a fertile ground for the spread of misinformation. From public health mis- and disinformation to political propaganda, unverified and often harmful content can quickly gain traction, influencing public opinions in significant ways. Moreover, misinformation generated by LLMs poses an additional risk to society, as they are able to generate convincing texts that can be potentially misused to spread mis- and disinformation [1, 2].

In response, automated fact-checking has become a vital tool in the fight against mis- and disinformation. However, an issue arises from the ability to extract and represent claims from noisy, informal and contextually ambiguous social media posts. They often lack clarity, use slang, and subjective or emotional language, which makes it difficult for the automated tools, but also for fact-checkers, to focus on the most important statements contained within the posts. This necessitates an intermediate

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain

† These authors contributed equally.

 tatiana.anikina@dfki.de (T. Anikina); ivan.vykopal@kinit.sk (I. Vykopal); sebastian.kula@kinit.sk (S. Kula); rach00004@teams.uni-saarland.de (R. K. Chikkala); natalia.skachkova@dfki.de (N. Skachkova); jing.yang@tu-berlin.de (J. Yang); veronika.solopova@tu-berlin.de (V. Solopova); vera.schmitt@tu-berlin.de (V. Schmitt); simon.ostermann@dfki.de (S. Ostermann)



© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

step – **claim normalization** – where unstructured and noisy social media posts are transformed into clear, concise, and verifiable claims. This process is crucial for extracting meaningful information from unstructured and cluttered posts, enabling more accurate and scalable fact-checking.

The global nature of false information highlights the importance of developing methods that are robust across languages. Deploying a unified approach for content moderation in multiple languages is not only more cost-effective, particularly for media organizations and journalists with limited computational resources, but also facilitates the identification and matching of related claims across different countries. In addition, the tools that are limited to a single language are insufficient in addressing the full scale of false information, making **multilingual claim normalization** essential for comprehensive fact-checking. To address these challenges, the *CLEF 2025 Shared Task on Multilingual Claim Extraction and Normalization* [3, 4, 5] focuses on simplifying and restructuring social media content by generating normalized claims. For instance, below is an example of a short social media post with the corresponding normalized claim:

Post: "A 40-ton truck lifted by 2,000 drones <https://t.co/lyBi5JNj7X> A 40-ton truck lifted by 2,000 drones <https://t.co/lyBi5JNj7X> A 40-ton truck lifted by 2,000 drones <https://t.co/lyBi5JNj7X> None."

Normalized Claim: "Thousands of drones lift a truck."

The shared task is organized into two settings: *monolingual* and *zero-shot*. The *monolingual setting* covers 13 languages, including both high and low-resource ones: *English, German, French, Spanish, Portuguese, Hindi, Marathi, Punjabi, Tamil, Arabic, Thai, Indonesian, and Polish*. This setting contains training, development and test data and thus enables model fine-tuning and language-specific evaluation when models are trained and tested on the data in the same language. *Zero-shot* is a more challenging setting that includes only the test data in 7 unseen languages – *Dutch, Romanian, Bengali, Telugu, Korean, Greek, and Czech*. The goal of this setting is to assess the generalization capabilities of LLMs without any language-specific training data.

We address the shared task by exploring various multilingual LLM-based approaches: zero-shot and few-shot prompting, LoRA adapters, and ensembling methods.² Based on the experimental results and our submissions to the shared task, we found that the best-performing approach largely depends on the language, the multilingual support of the LLM, and the amount of available data for fine-tuning and few-shot prompting. In the *zero-shot setting*, the best scores were achieved either with prompting a large multilingual Gemma3 27B model, or by using an ensemble of methods as described in Section 3.2.4 that combines the outputs of different approaches by selecting the most representative samples. In the *monolingual setting*, the best scores were obtained either with adapter-based fine-tuning (for 4 languages), few-shot prompting (3 languages), or with ensembling (6 languages). The ensemble method proved to be an overall very successful strategy for selecting the most appropriate normalized claims in our experiments.

2. Related Work

Multilingual Fact-Checking. Fact-checking is a multi-step process, typically involving claim detection, claim-matching, evidence retrieval and claim verification [6]. In multilingual contexts, the pipeline faces additional challenges due to the linguistic diversity and varying resource availability across languages. Previous work aimed to address this issue by extending the fact-checking datasets beyond English, with additional languages. Chang et al. [7] introduced a multilingual version of the FEVER dataset [8], a dataset constructed using machine translation into five additional languages. Other popular multilingual datasets include X-Fact [9] or MultiClaim [10], which focused on more diverse languages, including low-resource ones.

²Our code is available at: <https://github.com/tanikina/clef2-normalization>

Existing research for multilingual approaches mostly focused on two directions: (1) translating data into English and using monolingual models [11]; or (2) directly using multilingual models on the data, whether by fine-tuning or by developing novel approaches for multilingual fact-checking [12, 13]. Recent studies have explored the use of LLMs in multilingual fact-checking. Singhal et al. [14] evaluated the multilingual capabilities of LLMs across five diverse languages using various techniques. However, challenges remain, and performance on the low-resource languages is still suboptimal [15].

Verified Claim Retrieval. Verified claim retrieval, also known as claim-matching [16] or previously fact-checked claim retrieval [10], is one of the important tasks within the fact-checking process [17]. While the primary goal of verified claim retrieval is to determine whether a given claim has already been fact-checked based on a set of previously verified claims, there are also auxiliary tasks designed to enhance the performance on this task [18].

Since the spread of false information is a global phenomenon, it is necessary to check the fact-checked claims across languages and not only in English. Therefore, the first multilingual datasets for claim-matching were developed [16, 19]. Pikuliak et al. [10] introduced the largest multilingual dataset, which includes fact-checks in 39 languages and social media posts in 27 languages.

The most common approach for verified claim retrieval includes using text embedding models (TEMs) [20, 10, 21] or BM25 [22, 10] for the identification of similar claims based on a given input. However, since the multilingual datasets mostly contain social media posts, the retrieval phase faces several challenges. One of the main problems is that some social media posts are long, especially those from Facebook, which makes the retrieval using semantic similarity more challenging. Furthermore, social media posts can contain information unnecessary for the retrieval and fact verification, which can impact the performance for particular tasks.

Claim Normalization. Claim normalization, a task related to verified claim retrieval, aims to transform complex, unstructured and noisy claims or social media posts into concise, standalone and verifiable statements. This process enhances the efficiency of fact-checking by facilitating better verified claim retrieval, evidence retrieval and verification. Sundriyal et al. [3] defined the claim normalization as the task of simplifying the claim made in a social media post in a concise form.

Sundriyal et al. [18] introduced the claim normalization task, which focuses on decomposing complex and noisy social media posts into more straightforward and understandable forms, termed as normalized claims. They proposed CACN, a novel approach that leverages the chain-of-thought and few-shot demonstrations to produce normalized claims. Their experiments demonstrated that CACN outperforms several baselines. However, they limit their experiments to English social media posts and English fact-checking data only.

Ni et al. [23] addressed challenges in factual claim detection, including inconsistent definitions. In their work, they aimed to standardize the definition of factual claims to avoid misconceptions. The authors defined the factual claim as a statement that contains objectively verifiable facts without subjective opinions. In some of our approaches, we build upon this definition and use it as a characteristic of the normalized claims.

In addition, Metropolitansky and Larson [24] proposed a framework for evaluating claim extraction in the context of fact-checking. They introduced *Claimify*, an LLM-based claim extraction and demonstrated that it outperforms existing methods under their evaluation framework. While the claim normalization and claim extraction are different tasks, both aim to produce concise and verifiable claims. While normalization simplifies and clarifies existing claims from a given text, extraction identifies such claims from a broader context and usually decontextualizes them for further verification. Despite the differences, both share the goal of generating clear claims suitable for automated fact-checking.

Table 1

Dataset statistics for the claim normalization task.

Language	Arabic (ara)	Bengali (ben)	Czech (ces)	German (deu)	Greek (ell)	English (eng)	French (fra)	Hindi (hin)	Korean (kor)	Marathi (mar)
Train	470	0	0	386	0	11374	1174	1081	0	137
Dev	118	0	0	101	0	1171	147	50	0	50
Test	100	81	123	100	156	1285	148	100	274	100
Language	Indonesian (msa)	Dutch (nld)	Punjabi (pan)	Polish (pol)	Portuguese (por)	Romanian (ron)	Spanish (spa)	Tamil (tam)	Telugu (tel)	Thai (tha)
Train	540	0	445	163	1735	0	3458	102	0	244
Dev	137	0	50	41	223	0	439	50	0	61
Test	100	177	100	100	225	141	439	100	116	100

3. Methodology

3.1. Dataset

The dataset for the CheckThat 2025 task of extracting and normalizing social media posts includes 20 languages from diverse language families and scripts [3]. Table 1 presents the statistics for each language. The task provides the data in two settings: *monolingual* and *zero-shot*. In the *monolingual setting*, the data contain all three splits – train, development and test, while in the *zero-shot setting*, only the test split is provided. Importantly, the shared task data are imbalanced, even when training splits are available, their size substantially differs between the languages: from 102 samples in *Tamil* to 11374 samples in *English* (see Table 1).

Data Collection. The data are sourced from the Google Fact-check Explorer API³ and are extracted from the Claim Review Schema⁴. The Claim Review Schema contains the fact-checked claims paired with the posts they address through the corresponding fact-check. Finally, the data for the task consists of pairs of social media posts and fact-checked claims, which serve as the normalized claims for the specific post [3].

Data Pre-Processing. We found that for some languages in the *monolingual setting*, there was a substantial overlap between the samples in the training and development data (see Figure 1 for the claim overlap and Figure 12 in the Appendix for the post overlap). Therefore, we applied some pre-processing and filtered out all exact duplicates, ensuring that the training and development data are non-overlapping. We also found that some posts and claims have mixed languages, e.g., the post can be in *Hindi* but its normalized claim is in *English*. Even when languages are the same, some claims in the training data have very low similarity to the corresponding gold posts. This can happen, e.g., when the post is referring to some image or video, but those are not provided together with the textual inputs, and therefore it is impossible for the model to generate correct claims for such cases. We used SentenceTransformers⁵ [25] to measure the similarity between the claims and posts and filtered out all cases with a similarity score less than 0.05. For language detection, we employed the *fasttext-langdetect* library [26] and discarded the cases where either the post or the gold claim was in *English* while the expected target was another language. The statistics regarding the filtered training data can be found in Table 2.

Moreover, we experimented with additional filtering and normalization methods. We tested on the development set whether we can improve the results by removing excessive punctuation and normalizing the hashtags and URLs, i.e., extracting meaningful tokens from them, such as converting *#MasksDoNotWork* into *masks do not work*, or <https://www.technocracy.news/blaylock-face-masks-pose-serious-risks-to-the-healthy/> into <https://www.technocracy.news/blaylock face masks pose serious risks>

³<https://toolbox.google.com/factcheck/apis>

⁴<https://schema.org/ClaimReview>

⁵<https://huggingface.co/sentence-transformers paraphrase-multilingual-MiniLM-L12-v2>

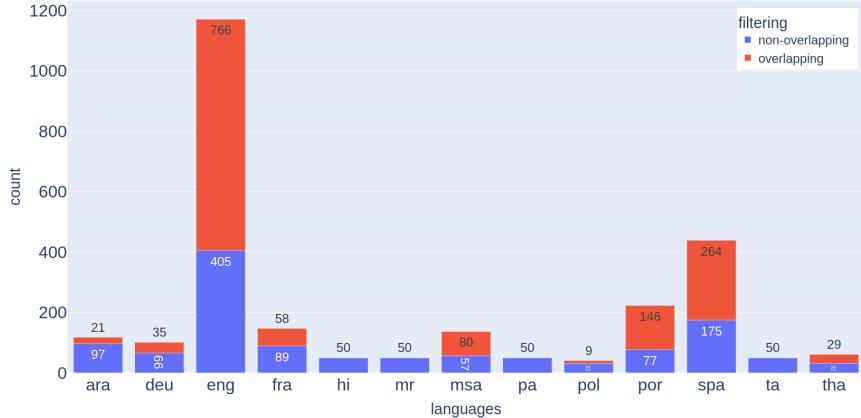


Figure 1: Claim overlap between the gold train and development data.

Table 2

Number of the filtered out samples per language. Numbers in **bold** indicate significant overlaps (more than 5%).

	ara	deu	eng	fra	hi	mr	msa	pa	pol	por	spa	ta	tha
Original train	470	386	11374	1174	1081	137	540	445	163	1735	3458	102	244
Filtered train	462	337	9342	1132	1048	128	520	423	151	1551	3288	101	241
% Filtered out	1.70	12.69	17.87	3.58	3.05	6.57	3.70	4.94	7.36	10.61	4.92	0.98	1.23

to the healthy. We also tried removing repeated text sequences in posts (see example in Section 1). However, cleaning the data in this way and using the “normalized posts” for prompting did not result in any substantial improvement of the final performance. Therefore, we only performed de-duplication and similarity filtering as described above and did not modify the original posts.

3.2. Experimental Setup

To perform the normalization of the social media posts, we experimented with various strategies and LLMs. Specifically, we focused on model fine-tuning with LoRA adapters and the prompting experiments. For evaluating the performance of the proposed methods, we leveraged the METEOR Score. We evaluated the final performance using the development sets for particular languages in the *monolingual setting*. In addition, we provide the results on test sets from the submitted results for both *monolingual and zero-shot settings*.

In this section, we describe the models used in our experiments (Section 3.2.1), fine-tuning of selected LLMs (Section 3.2.2) and prompting experiments with various scenarios (Section 3.2.3).

3.2.1. Models

For our experiments and the proposed methods, we selected multiple LLMs, which are detailed in Table 3. Specifically, we focused on multilingual LLMs with various model sizes ranging from 8B to 405B and compared their efficiency in generating normalized claims.

In total, we employed 9 LLMs in various experiments, especially focusing on parameter-efficient fine-tuning and prompting. Most of these LLMs were used primarily for prompting experiments across all languages or particular experiments for the Polish language. Additionally, Gemma3 4B, Gemma3 27B, and Qwen3 14B were fine-tuned using LoRA adapters to further tailor their performance to the claim normalization task.

Table 3

A list of LLMs used in our experiments with the indicator, which approaches were employed, whether fine-tuning the LoRA adapter or prompting techniques.

Model	# Params	# Langs	Citation	LoRA	Prompting
Llama3.1 Instruct	405 B	8	Grattafiori et al. [27]		✓
Llama3.1 Nemotron Ultra	253 B	8	Bercovich et al. [28]		✓
Qwen2.5 Instruct	72 B	29	Yang et al. [29]		✓
Llama3.3 Instruct	70 B	8	Grattafiori et al. [27]		✓
Qwen3	32 B	100+	Yang et al. [30]		✓
Gemma3 IT	27 B	140+	Team et al. [31]	✓	✓
Qwen3	14 B	100+	Yang et al. [30]	✓	
Bielik Instruct v2.3	11 B	1	Ociepa et al. [32]		✓
Qwen3	8 B	100+	Yang et al. [30]		✓

3.2.2. Parameter-Efficient Fine-Tuning

For the *monolingual setting*, we fine-tuned LoRA adapters [33] for the Qwen3 14B model⁶ using the Unsloth library⁷. In addition, we experimented with fine-tuning Gemma3 4B and Gemma3 27B. However, based on the performance on the development set, we chose Qwen3 14B for the shared task submission. We also experimented with both short and verbose task descriptions as additional input to the model and found that the verbose version results in better METEOR scores. This verbose version provides a detailed task description and the definition of the normalized claim with the criteria based on [18], we used this version for all adapter-based submissions. More details regarding the adapter fine-tuning, including the hyperparameter values, can be found in Appendix B.

We also checked whether the generated claim is a valid text, because sometimes LLM generates a long string of repeated characters or tokens. To avoid such nonsensical outputs, we checked whether the output claim contained less than three different tokens or less than five different characters and repeated generation if this was the case. We also set a constraint that the output should not contain *http* because this is an indicator that some URLs were copied from the post, which typically results in badly normalized claims.

3.2.3. Prompting Experiments

In this section, we describe several experiments for the *monolingual and zero-shot settings* across languages. We divided these experiments into two categories: (1) *monolingual and zero-shot experiments*, where we experimented with LLMs across all 20 languages within the shared task; and (2) *Polish experiments*, in which we experimented with LLMs particularly only for the Polish language and also with one Polish LLM – Bielik Instruct v2.3.

Furthermore, we performed additional prompting experiments using Direct and Summarization based normalization techniques for both *monolingual* and *zero-shot settings* across languages, see section C.3 in the Appendix.

Monolingual and Zero-Shot Experiments. Given that the claim normalization task also includes *zero-shot setting*, where the training and development data are not available, we experimented with various prompting techniques to address this limitation. Specifically, we experimented with: (1) *zero-shot prompting*; (2) *few-shot prompting* with a varied number of demonstrations; (3) *translated zero-shot prompting*; and (4) *translated few-shot prompting*. In addition, for the few-shot prompting and translated few-shot prompting, we experimented with using the filtered and unfiltered data for selecting demonstrations for the prompt. In our experiments with LLMs, we set *do_sample=False* to enforce greedy decoding, ensuring deterministic output by selecting the most probable next token.

⁶<https://huggingface.co/unsloth/Qwen3-14B>

⁷<https://github.com/unslothai/unsloth>

In ***Zero-Shot prompting***, we provide LLMs with the task description and the main characteristics that the normalized claims should fulfill. In this scenario, we rely on the LLM’s understanding of the task based on the given instructions in English without any previous examples (see Figure 7). For the ***Translated Zero-Shot prompting***, we utilized Google Translate for translating the English prompt into particular languages for both *monolingual* and *zero-shot settings*.

The characteristics of the normalized claim can be complex to comprehend, and there are variances across languages in what the normalized claims look like. Therefore, we employed the ***Few-Shot prompting***, in which we extended the zero-shot prompting by providing demonstrations from the training data, while the instruction is in English (see Figure 8). In the ***Translated Few-Shot prompting***, we translated the instruction into particular languages, while the demonstrations are kept in the original languages as sampled from the training set.

To select few-shot demonstrations, we utilized the semantic similarity between posts using the GTE-Multilingual-Base⁸ [34] embedding model, which supports more than 70 languages. We calculated the similarity between the analyzed social media post and the posts that are contained in the training data and selected the top K as demonstrations. We experimented with prompts containing 1, 2, 5 and 10 demonstrations. For few-shot prompting, we selected the most similar samples across all languages and not only from the particular language. Given the fact that there are languages for which we do not have any training data, we decided to select samples from the combined training set of all languages.

Few-shot experiments were done in two variants: *filtered* and *unfiltered*. In the filtered scenario, we used the filtered training data for selecting demonstrations, in which we removed the posts that were included in various splits for a particular language and not only in the training set as described in Section 3.1. For the unfiltered scenario, we employed original training sets and especially the combination of all training data for the sample selection process.

Polish Experiments. In our additional experiments, we specifically focused on Polish, a low-resource language, which consists of a total of 304 samples. The limited size of this dataset motivated a more comprehensive analysis of the application of various LLMs and diverse prompts to achieve performance comparable to that of models for high-resource languages, such as *English* and *French*. Specifically, we selected *Polish* over other low-resource languages, such as *Tamil* or *Marathi*, to focus on Latin-script languages and reduce variability from different writing systems. Polish also represents the unrepresented Slavic language family in multilingual NLP, allowing us to address this gap. Furthermore, having a native Polish speaker among the authors enabled more accurate evaluation and interpretation of LLM outputs.

For experiments with the Polish language, we employed three LLMs, especially the *Bielik v2.3* – Polish model and multilingual *Llama3.1 Nemotron Ultra* and *Llama3.1 405B*. For these LLMs, we leveraged two prompting strategies: (1) *Chain-of-Thought* (CoT) and (2) *Few-Shot prompting*.

The CoT prompt in *Polish* was developed with the assistance of the *Llama3.1 405B* model and relevant research papers, especially by Sundriyal et al. [18] and Sundriyal et al. [3]. We instructed the *Llama3.1 405B* model to generate a CoT prompt based on the description of the task and the normalized claim. We refer to this prompting strategy as *Polish-CoT*, which is shown along with the English translation in Figure 9 in the Appendix. These experiments with *Polish-CoT* were done only for *Bielik v2.3*.

The second set of experiments investigated the effectiveness of a few-shot strategy, specifically using 3, 10, and 20-shot prompting. For few-shot prompting, we selected demonstrations from the unfiltered training set based on a cosine similarity using the *paraphrase-multilingual-MiniLM-L12-v2* model. An example of a system prompt and a few-shot prompt used can be found in Figure 10 and Figure 11 in the Appendix.

⁸<https://huggingface.co/Alibaba-NLP/gte-multilingual-base>

3.2.4. Ensemble of Methods

For method ensembling, we first collected the data from the five top-performing generation strategies (the exact setting depends on the language, and may include few-shot prompting with different models and fine-tuned LoRA adapters). Second, we compute a centroid (averaged) embedding for each normalized claim based on the sentences encoded with `paraphrase-multilingual-MiniLM-L12-v2` model. Third, we computed the similarity score between all claims generated by the top-5 methods for the same post and their centroid embedding, and selected as the final output the claim that has the highest similarity to the centroid. The idea behind this approach is to leverage the “wisdom of the crowd” and find the most common representation of the generated claims. LLM outputs may differ in quality depending on the input, for instance, sometimes the claim is generated in the wrong language or includes some hallucinated content, but if in 4 out of 5 cases the generated claim uses the correct target language and references the same core content, this issue will be self-corrected by automatically picking the most representative sample with the embedding closest to the centroid.

4. Evaluation

In this section, we present our findings on parameter-efficient fine-tuning and LLM prompting for the claim normalization task. We begin with the results from the *monolingual setting* (Section 4.1), including observations from LoRA fine-tuning (Section 4.1.1) and evaluations of various prompting techniques (Section 4.1.2). Additionally, we report the final results from the shared task submission platform (Section 4.2), covering both the *monolingual* and *zero-shot settings*. This includes ranking of our methods and the identification of the best-performing approaches for specific languages on the test set.

4.1. Monolingual Settings

In the *monolingual setting*, we evaluate our approaches by training and testing on the data from the same languages. This allows us to focus on language-specific performance and assess the effectiveness of parameter-efficient fine-tuning and prompting methods. Since ground truth labels for the test set are unavailable, we report the final performance of our approaches based on the development set.

4.1.1. Parameter-Efficient Fine-Tuning Results

Based on the initial prompting results, we found that the multilingual Gemma3 27B model achieves good results for many languages in the *monolingual setting*. Therefore, we focused on that model when doing experiments with LoRA adapters (see Table 4), but replaced Gemma3 27B with Qwen3 14B for the final submission, because Qwen3 outperformed Gemma3 and showed the best average performance in our later experiments with prompting (see Section 4.1.2 for more detail). We did not repeat the same experiments with Qwen3 due to the lack of time and computational resources, and directly fine-tuned the adapters on the de-duplicated training set prepared according to Section 3.1.

Given that the shared task data are imbalanced (see Table 1), we experimented with different ways of augmenting and balancing the data to mitigate this issue. For instance, LoRA-translated in Table 4 relies on data augmentation via translation from English into the target languages. We used the Google Translate API and selected the posts with less than 1500 characters as source data. The translated posts and normalized claims were then combined with the original samples and used for fine-tuning the adapters (see Appendix B for the fine-tuning details). We also experimented with filtering out “bad translations” by applying a set of heuristics (*LoRA-translated-v2* in Table 4). In this setting, we ensure that both social media posts and claims share the same target language, and the cosine similarity between each translated post and the corresponding gold claim is above the median computed on the train data for each language using Sentence Transformer model `paraphrase-multilingual-MiniLM-L12-v2`.

To mitigate the imbalance without adding new data points, we also considered the setting, where we fine-tune a single adapter on the mixed data from different languages, *LoRA-all-balanced* in Table 4, but

all post-claim pairs are subsampled to 500 per language to ensure equal representation and diversity. Since the gains in performance were marginal, we did not repeat these experiments for Qwen3 14B and used the original, non-translated data, training a separate adapter for each language.

Table 4

Fine-tuning results for Gemma3 27B evaluated on the official development set. The first row shows the zero-shot prompting results for comparison.

Approach	ara	deu	eng	fra	hi	mr	msa	pa	pol	por	spa	ta	tha
Zero-shot	0.305	0.161	0.244	0.265	0.224	0.275	0.219	0.311	0.194	0.294	0.268	0.340	0.054
LoRA-target	0.361	0.298	0.658	0.439	0.290	0.311	0.599	0.352	0.267	0.509	0.518	0.450	0.217
LoRA-all-balanced	0.390	0.293	N/A	0.454	0.285	0.287	0.570	0.309	0.265	0.510	0.531	0.438	0.213
LoRA-translated	0.379	0.302	N/A	0.430	0.254	0.297	0.551	0.290	0.236	0.497	0.509	0.332	0.175
LoRA-translated-v2	0.369	0.280	N/A	0.420	0.283	0.286	0.623	0.313	0.315	0.504	0.535	0.457	0.198

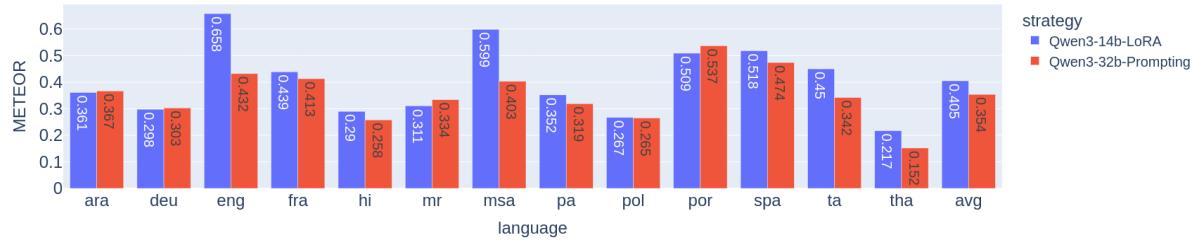


Figure 2: Adapter tuning vs. few-shot prompting for Qwen3 on the development set (based on the filtered data).

The results in Table 4 (based on the development data) indicate that **the basic LoRA adapter separately fine-tuned on each target language (LoRA-target) already achieves the optimal performance** for *English, Hindi, Marathi, Punjabi* and *Thai*. Using a single adapter fine-tuned on the mixture of different languages with roughly equal representation (LoRA-balanced) results in small improvements for *Arabic, French, and Portuguese*, and using translated data without any additional filtering (LoRA-translated) is slightly beneficial only for *German*. Note that filtering out bad examples from the training set and ensuring high similarity between the translated claims and posts that have the correct target language (LoRA-translated-v2) is beneficial for some languages and namely leads to small improvements for *Indonesian, Polish, Spanish, and Tamil*. However, due to the fact that fine-tuning adapters on a large amount of translated data is computationally expensive and brings only marginal gains, we decided to fine-tune the Qwen3 adapters only on the original data for each language. The final results, including the fine-tuned adapters and the ensemble method, are discussed in more detail in Section 4.2.

The comparison between the fine-tuned LoRA adapters with Qwen3 14B and the few-shot prompting of Qwen3 32B (best strategy according to Table 5 with filtered data) is shown in Figure 2.⁹ **The results indicate that for some languages** (e.g. *German, Polish, Arabic*) **the difference in performance is negligible**, while for others (e.g. *Indonesian* and *English*) adapters substantially outperform few-shot prompting. Although the amount of training data has some impact on the downstream performance (as indicated by much better performance on the *English* data), the pattern is not consistent. For instance, *Portuguese* has more than 1500 samples in the training set, but few-shot prompting outperforms adapters, while *Tamil* has only 100 samples, but adapters achieve the best METEOR score (+10.8% compared to the few-shot prompting).

Overall, **high-resource languages with a significant amount of training data** (*English, Spanish, Portuguese, and French*) **demonstrate relatively good performance** (0.44-0.65), and when high- or mid-resource languages have comparatively less data (<500 for *German* and *Arabic*), they tend to underperform (0.30-0.36). As for the low-resource languages, adapters work well for *Tamil* and *Punjabi*

⁹We did not fine-tune adapters for Qwen3 32B because of the limited computational resources at the time of the submission.

Table 5

LLM performance in the *monolingual setting* on the development set. *En* indicates prompts written in English, while *Og* refers to prompts translated into the target language (e.g., *Arabic* prompts for the *ara* language). The *Fil.* column specifies the few-shot prompting setup: ✓ denotes that filtered data was used to sample demonstrations, whereas an empty cell indicates the use of unfiltered data. Best results for each language are in **bold** and second-best are underlined.

Model	Technique	Vs.	Fil.	ara	deu	eng	fra	hi	mr	msa	pa	pol	por	spa	ta	tha	Avg.
Qwen3 (8B)	Zero-Shot	En		0.361	0.142	0.248	0.253	0.251	0.260	0.199	0.318	0.181	0.275	0.258	<u>0.412</u>	0.060	0.247
	Zero-Shot	Og		0.270	0.143	0.248	0.260	0.211	0.217	0.218	0.284	0.176	0.276	0.270	0.333	0.024	0.225
	10-Shot	En		0.355	0.241	0.550	0.373	0.234	0.309	0.361	0.341	0.237	0.460	0.428	0.390	0.192	0.344
	10-Shot	En	✓	0.342	0.244	0.432	0.375	0.264	0.333	0.345	0.331	0.224	0.463	0.445	<u>0.412</u>	0.175	0.337
	10-Shot	Og		0.377	0.248	0.550	0.377	0.219	0.303	0.386	0.361	0.262	0.494	0.423	0.350	0.166	0.347
	10-Shot	Og	✓	0.367	0.228	0.432	0.380	0.255	0.313	0.366	<u>0.351</u>	0.235	0.466	0.408	0.365	0.139	0.331
Gemma3 (27B)	Zero-Shot	En		0.305	0.161	0.244	0.265	0.224	0.275	0.219	0.311	0.194	0.294	0.268	0.340	0.054	0.243
	Zero-Shot	Og		0.315	0.159	0.244	0.262	0.239	0.149	0.219	0.251	0.189	0.283	0.271	0.294	0.045	0.225
	10-Shot	En		0.312	0.280	0.479	0.371	0.231	0.308	0.404	0.310	0.236	0.472	0.428	0.349	0.222	0.339
	10-Shot	En	✓	0.306	0.261	0.357	0.368	0.226	0.327	0.364	0.314	0.241	0.468	0.413	0.348	0.196	0.322
	10-Shot	Og		0.316	0.292	0.479	0.387	0.249	0.295	0.429	0.331	0.229	0.506	0.454	0.387	0.283	0.357
	10-Shot	Og	✓	0.316	0.268	0.357	0.384	0.258	0.265	0.414	0.326	0.225	0.473	0.427	0.376	0.208	0.330
Qwen3 (32B)	Zero-Shot	En		0.376	0.154	0.258	0.276	<u>0.275</u>	0.320	0.220	0.342	0.218	0.323	0.283	0.351	0.054	0.265
	Zero-Shot	Og		0.284	0.173	0.258	0.271	0.220	0.215	0.217	0.262	0.194	0.295	0.282	0.206	0.037	0.224
	10-Shot	En		0.348	<u>0.271</u>	0.587	0.409	0.293	0.324	0.420	0.350	0.224	0.536	<u>0.485</u>	0.416	0.211	0.375
	10-Shot	En	✓	0.367	0.303	0.432	0.413	0.258	<u>0.334</u>	0.403	0.319	0.265	0.537	0.474	0.342	0.152	0.354
	10-Shot	Og		0.330	0.278	0.585	0.411	0.271	0.248	<u>0.441</u>	0.329	0.231	0.551	0.478	0.345	0.190	0.360
	10-Shot	Og	✓	0.366	0.328	0.432	0.419	0.264	0.245	0.399	0.326	0.284	<u>0.549</u>	0.461	0.326	0.152	0.350
Llama3.3 (70B)	Zero-Shot	En		0.358	0.141	0.248	0.249	0.242	0.285	0.188	0.337	0.205	0.295	0.273	0.337	0.059	0.247
	Zero-Shot	Og		0.312	0.163	0.248	0.267	0.226	0.139	0.210	0.182	0.212	0.280	0.267	0.333	0.056	0.223
	10-Shot	En		0.249	0.242	0.481	0.349	0.196	0.331	0.365	0.333	0.254	0.476	0.432	0.370	0.170	0.327
	10-Shot	En	✓	0.218	0.229	0.372	0.345	0.206	0.341	0.353	0.326	0.220	0.447	0.405	0.290	0.150	0.300
	10-Shot	Og		0.300	0.272	0.481	0.395	0.227	0.215	0.467	0.105	0.292	0.494	0.447	0.191	0.189	0.314
	10-Shot	Og	✓	0.313	0.271	0.372	0.361	0.221	0.213	0.436	0.137	0.278	0.481	0.439	0.221	0.167	0.301
Qwen2.5 (72B)	Zero-Shot	En		0.344	0.134	0.247	0.261	0.226	0.312	0.203	0.318	0.164	0.274	0.253	0.302	0.074	0.240
	Zero-Shot	Og		0.302	0.160	0.246	0.274	0.227	0.193	0.212	0.248	0.175	0.289	0.289	0.277	0.032	0.225
	10-Shot	En		0.335	0.266	0.514	0.379	0.176	0.313	0.356	0.270	<u>0.284</u>	0.521	0.460	0.329	0.260	0.343
	10-Shot	En	✓	0.317	0.241	0.398	0.362	0.208	0.308	0.343	0.334	0.248	0.485	0.438	0.344	0.194	0.325
	10-Shot	Og		0.321	0.269	0.514	0.400	0.225	0.276	0.427	0.251	0.261	0.525	0.486	0.386	<u>0.276</u>	0.355
	10-Shot	Og	✓	0.343	0.254	0.398	0.399	0.219	0.262	0.404	0.275	0.232	0.494	0.461	0.339	0.211	0.330

but achieve slightly worse results for *Marathi*. Both methods obtain almost identical scores for *Polish* that has a very small amount of training data (only 151 samples after filtering). On average, languages with non-Latin script (*Arabic*, *Hindi*, *Marathi*, *Punjabi*, *Tamil*, and *Thai*) obtain lower scores than the ones with Latin script (0.33 vs. 0.47).

4.1.2. Prompting Experiments

In addition to LoRA adapters, we employed various strategies for instructing LLMs with a specific focus on evaluating the results of the proposed approaches in the *zero-shot setting*, where we are not provided with the training and development sets.

Zero and Few-Shot Prompting. For the comprehensive evaluation of various settings across languages in the *monolingual settings*, we evaluated the zero-shot and few-shot prompting along with the translated version. The overall results are shown in Table 5, where we provide the results across 13 languages, five LLMs and in six settings. In addition, we compare the prompts written in English versus those written in the target language to measure the impact of the instruction language on the model’s performance.

Across all models, we observe a consistent improvement when moving from zero-shot to few-shot prompting. **The best average performance in the monolingual setting is achieved by Qwen3 32B in the 10-shot setting with the English instruction** and when using unfiltered data for selecting samples. In addition, Gemma 3 27B performed comparably well when using 10-shot prompting with the instruction in the target language without unfiltered data.

In zero-shot prompting, **the prompts written in English consistently outperformed those in the target language**, which can be caused by the fact that since LLMs are trained on a variety of

Table 6

LLM performance on the Polish language. The best results on the test set were achieved by Llama3.1 405B using 20 samples in the prompt.

Model	Prompt Type	Dev Set	Test Set
Bielik Instuct v2.3	Polish-CoT	0.198	N/A
	3-shot	0.282	0.297
Llama3.1 Nemotron Ultra	3-shot	0.254	N/A
	10-shot	0.296	0.347
Llama3.1 405B	10-shot	0.271	0.393
	20-shot	N/A	0.396

languages, English still presents the major part of the pre-training, and therefore, the LLM can still better process input when using English instruction instead of translated instructions. However, in a few-shot prompting, prompts in the target language (Og) outperformed those in English across most languages, specifically for Gemma3 and Qwen2.5 models, suggesting that aligning the instruction language with the input language and demonstrations helps the model better contextualize the task.

High-resource Western European languages, such as *Spanish, English, French, Portuguese*, demonstrated consistently strong performance, with *English* and *Portuguese* achieving the highest scores, both exceeding 0.55. In addition, as can be expected, ***English showed the strongest performance across many LLMs***, particularly using the 10-shot setting. Notably, Qwen3 32B reaches the highest score of 0.59, indicating the model’s strong performance in *English*.

Languages with non-Latin scripts, such as *Arabic, Thai, Tamil, Hindi, Marathi*, and *Punjabi*, showed more variable performance. Among them, ***Arabic and Tamil performed the best with Qwen3 models*** (whether 8B or 32B). On the other hand, *Thai* achieved relatively low performance, especially using zero-shot prompting, with a maximum 0.07 METEOR score. However, by providing demonstrations, the performance increased to more than 0.28.

Surprisingly, the *German* language exhibited very low performance across LLMs, particularly using zero-shot prompting. This outcome may be attributed to issues with data quality, as our manual inspection revealed several issues. In some cases, the normalized claims associated with social media posts were written in a different language, or the key information from the normalized claim was absent from the post. Such discrepancies likely hinder the model’s ability to generate appropriate claims, especially without additional context. Moreover, many normalized claims referenced images or videos that were not included in the input. As a result, LLMs were not able to recognize or indicate that certain claims were grounded in visual evidence.

Prompting Results for Polish. For the *Polish* language, we conducted a separate set of experiments and evaluated it on the development set, where the samples from the unfiltered training set were employed as demonstrations for few-shot prompting. In addition, we provide the results on the test set obtained from the submission site, where both training and development sets were used for demonstration selection.

The results from Table 6 indicate that the optimal performance for *Polish* on the development dataset was achieved using the *Llama3.1 Nemotron Ultra* model with a 10-shot learning approach. In contrast, the best results on the test dataset were obtained using the *Llama3.1 405B* model with a 20-shot learning approach.

4.2. Final Results

Table 7 presents the final evaluation results of our proposed approaches on the official test set for the shared task, covering both *monolingual* and *zero-shot settings*. Our approaches performed competitively across a wide range of languages, **achieving the first rank in 13 out of the 20 evaluated languages**.

Table 7Final evaluation results on the official test set for both monolingual and zero-shot (indicated as *zero*) settings.

Language	Arabic (ara)	German (deu)	English (eng)	French (fra)	Hindi (hi)	Marathi (mr)	Indonesian (msa)	Punjabi (pa)	Polish (pol)	Portuguese (por)
Best Score	0.504	0.386	0.457	0.527	0.328	0.389	0.565	0.331	0.407	0.577
Our Score	0.504	0.347	0.457	0.470	0.328	0.389	0.502	0.331	0.396	0.574
Δ (Ours vs Best)	0	-0.039	0	-0.057	0	0	-0.063	0	-0.011	-0.003
Our Strategy	Ensemble	Qwen3-32b	Ensemble	Qwen3 _{LoRA}	Ensemble	Qwen3 _{LoRA}	Qwen3 _{LoRA}	Qwen3-8b	Llama3.1	Ensemble
Our Rank	1	2	1	2	1	1	2	1	2	2
Language	Spanish (spa)	Tamil (ta)	Thai (tha)	Bengali _{zero} (ben)	Czech _{zero} (ces)	Greek _{zero} (ell)	Korean _{zero} (kor)	Dutch _{zero} (ndl)	Romanian _{zero} (ron)	Telugu _{zero} (te)
Best Score	0.608	0.632	0.586	0.378	0.252	0.262	0.134	0.200	0.295	0.526
Our Score	0.554	0.632	0.300	0.378	0.252	0.262	0.134	0.200	0.295	0.526
Δ (Ours vs Best)	-0.054	0	-0.286	0	0	0	0	0	0	0
Our Strategy	Ensemble	Qwen3 _{LoRA}	Ensemble	Ensemble	Gemma3	Ensemble	Gemma3	Ensemble	Ensemble	Ensemble
Our Rank	2	1	3	1	1	1	1	1	1	1

In the *monolingual setting*, we achieved the top score in six languages, especially *Arabic*, *English*, *Hindi*, *Marathi*, *Punjabi* and *Tamil*. Notably, from our proposed approaches, **the ensemble methods performed the best for six languages**, while fine-tuned LoRA adapters for Qwen3 model achieved superior performance on four languages. **Fine-tuned Qwen3 demonstrated strong performance in low-resource scenarios** like *Marathi*, *Indonesian* or *Tamil*. In addition, prompting techniques with Qwen3 shown to be effective for *German* and *Punjabi*.

In the *zero-shot setting*, **our methods obtained the highest score in all seven languages**. This demonstrated the generalization capabilities of our approaches even in the absence of training data for the target languages. Here, the use of Gemma3 and the ensemble of methods were crucial for achieving the best performance.

The largest gap between our score and the overall best score occurred for *Thai*, where our ensemble of methods scored 0.30 against the best of 0.59, placing us third. This suggests a potential area for improvement that involves exploring further prompting strategies and model adaptation in syntactically diverse languages.

5. Discussion

Our Main Findings. Our experiments show that LLMs are capable of performing the task of claim normalization for a variety of languages even when no or only few samples are available. However, different models may generate claims of different quality for the same post. Therefore, it is important to further “normalize” and post-process generated claims by using the ensemble method to find the most representative sample for each claim. This method resulted in the best score for 5 out of 7 languages in the *zero-shot setting* of the shared task, and it was the best strategy for almost half of the languages in the *monolingual setting*.

Overall, LLMs like Gemma3 and Qwen3 demonstrate strong multilingual capabilities. Gemma3 turned out to be the strongest model for *Czech* and *Korean* in the *zero-shot setting*, while Qwen3 showed better performance in the *monolingual setting*. Models of larger sizes (e.g., 32 B for Qwen and 27 B for Gemma) are generally better at claim normalization, but for some configurations and languages smaller models perform on-par or even outperform the larger ones. Additionally, we found that extra pre-processing and cleaning of the data does not substantially improve the scores, and our best results, depending on the language, were achieved with a few-shot prompting or fine-tuning with the original data.

Limitations & Challenges. The shared task presents several challenges and limitations. The provided dataset is unbalanced, and for some languages, there are thousands of examples (*English*, *Spanish*) while for others, it is only a few hundreds (*Polish*, *Marathi*, *Tamil*). Languages have different scripts, and some of them are very low-resource (e.g., *Bengali*, *Punjabi*, and *Telugu*).

A key limitation concerns the post and claim overlap across the dataset splits. While we identified and addressed the problem of overlapping claims and posts between the original training and development

data, the potential overlap between the training and testing data has not been analyzed. This makes the fine-tuning and few-shot prompting somewhat unreliable unless all overlapping instances are removed. This issue introduces an evaluation bias, especially in monolingual settings, where the models may appear to perform better due to memorization rather than generalization.

The input lengths can vary significantly, and some posts are very long and exceed the context window of LLMs, requiring truncation (posts can be up to 31843 characters and 5020 tokens in the *English* training set). Posts often include a lot of repetition along with excessive punctuation, emojis, URLs, hashtags, and ungrammatical sentences. Some gold posts and claims also appear in different languages, adding complexity to both fine-tuning and the interpretation of demonstrations. A number of claims also reference external media, such as videos or images, which are not included in the input, and this leads to potential loss of context and incorrectly or incompletely generated claims.

In addition, some social media posts include language that can be offensive, and LLMs refuse to generate any normalized claims based on such content, e.g., *“I understand you’ve expressed strong negative feelings and used offensive language towards Greta Thunberg. I want to be clear that I cannot and will not generate responses that include hate speech, insults, or profanity. My purpose is to be helpful and harmless, and that includes respecting individuals regardless of differing opinions.”* Although the models were instructed to act as fact-checkers or experts in detecting misinformation, they still refused to generate normalized claims in certain cases. This behaviour, however, also demonstrates their ability to refuse potentially harmful content and further spread misinformation.

Furthermore, understanding certain claims may require world knowledge or familiarity with specific events. The lack of context is an important limitation of the shared task data because some of the posts cannot be normalized without access to the conversational threads and additional media accompanying the post. E.g., it is not possible to infer “girl” in the gold normalized claim “*Girl from Ethiopia’s Mursi tribe*” based solely on “*Mursi tribe Ethiopia Africa Mursi tribe Ethiopia Africa Mursi tribe Ethiopia Africa None*”. Therefore, gold annotations are not always a realistic goal for the generated output, and having such examples in the gold data may encourage model hallucinations.

Future Work. In the future, researchers can consider experimenting with different approaches to data augmentation. For instance, LLMs can be leveraged to generate more samples (post-claim pairs) for underrepresented languages, and such data could then be further used for adapter fine-tuning. In addition, we see the potential in refining model predictions by applying self-revision, and the ensemble method that proved to be successful in our experiments could be applied to the outputs of the same model (i.e., one could do self-ensemble and find the most representative claims among all generated variants). Although both self-ensemble and self-revision increase inference time, they have the potential to improve the quality of generated data and avoid outliers, which is very important for low-resource scenarios.

Another direction to pursue is to test different ways of integrating additional constraints in the prompt and performing checks after the generation (e.g., ensuring that both the post and its normalized claim have the same language, and their similarity score is above the threshold derived based on the training data). We used some of the constraints when generating claims with adapters, but not in the prompting experiments. One could also benchmark additional multilingual models (e.g., Aya-100 [35]) and use soft prompts instead of adapters for parameter-efficient fine-tuning. Furthermore, the integration of dynamic selection of in-context demonstrations without relying on a fixed number of samples (top K) can be investigated in future work. This is especially important for the languages that do not have much data that can be used as demonstrations. The selection of the demonstrations based on the similarity threshold can help to eliminate those examples that could potentially harm the performance.

In addition, future work could explore the impact of using normalized claims on other fact-checking tasks, such as claim-matching, evidence retrieval, or fact verification. There are already efforts to evaluate the impact of claim decomposition on the fact-checking performance [36]. However, the effect of normalized claims on the performance of particular tasks has not been analyzed. Claims normalization may help reduce noise and ambiguity, potentially leading to improved model performance on these

tasks. Especially, in claim-matching, normalized claims can enhance the identification of whether a given claim was previously fact-checked, since normalized claims more closely resemble the statements with which they are being compared in this task, e.g., when using semantic similarity. A comparative analysis between using raw social media posts and their corresponding normalized claims would provide valuable insights into the benefits and limitations of normalization. Moreover, it would be interesting to conduct a feasibility analysis in real time settings by integrating multilingual LLM-based claim normalization in fact-checking workflows and see how this approach can be scaled.

6. Conclusion

In this paper, we presented our approaches to multilingual claim normalization in the context of the CLEF 2025 CheckThat! shared task. By combining parameter-efficient fine-tuning, prompting strategies, and ensemble methods, we addressed the challenges posed by noisy, informal, and multilingual social media content. Our methods demonstrated strong performance across both *monolingual* and *zero-shot settings*, achieving first place in 6 out of 13 *monolingual* languages and top scores in all 7 *zero-shot* languages.

We found that the effectiveness of each approach varied by language and resource availability. LoRA-based fine-tuning proved effective for low-resource scenarios, while few-shot prompting with models like Qwen3 32B yielded the best results in high-resource settings. The ensemble method, leveraging outputs from multiple strategies, emerged as a robust solution for selecting representative normalized claims, especially in *zero-shot* scenarios. Our findings highlight the potential of multilingual LLMs for claim normalization and their adaptability across diverse languages.

Acknowledgments

This research was partially supported by *DisAI - Improving scientific excellence and creativity in combating disinformation with artificial intelligence and language technologies*, a project funded by Horizon Europe under GA No.101079164, by *LorAI - Low Resource Artificial Intelligence*, a project funded by Horizon Europe under GA No.101136646, by the Ministry of Education, Youth and Sports of the Czech Republic through the e-INFRA CZ (ID:90254), by the German Federal Ministry of Research, Technology and Space (BMFTR) as part of the projects TRAILS (01IW24005) and VeraExtract (01IS24066), as well as BIFOLD Agility Project FakeXplain.

Declaration of Generative AI

During the preparation of this work, some authors used Grammarly and ChatGPT in order to check the spelling and paraphrase. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

- [1] I. Vykopal, M. Pikuliak, I. Srba, R. Moro, D. Macko, M. Bielikova, Disinformation Capabilities of Large Language Models, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 14830–14847. URL: <https://aclanthology.org/2024.acl-long.793/>. doi:10.18653/v1/2024.acl-long.793.
- [2] A. Zugecova, D. Macko, I. Srba, R. Moro, J. Kopal, K. Marcincinova, M. Mesarcik, Evaluation of LLM Vulnerabilities to Being Misused for Personalized Disinformation Generation, 2024. URL: <https://arxiv.org/abs/2412.13666>. arXiv:2412.13666.

- [3] M. Sundriyal, T. Chakraborty, P. Nakov, Overview of the CLEF-2025 CheckThat! lab task 2 on claim normalization, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CLEF 2025, Madrid, Spain, 2025.
- [4] F. Alam, J. M. Struß, T. Chakraborty, S. Dietze, S. Hafid, K. Korre, A. Muti, P. Nakov, F. Ruggeri, S. Schellhammer, V. Setty, M. Sundriyal, K. Todorov, V. V., The clef-2025 checkthat! lab: Subjectivity, fact-checking, claim normalization, and retrieval, in: C. Hauff, C. Macdonald, D. Jannach, G. Kazai, F. M. Nardini, F. Pinelli, F. Silvestri, N. Tonellootto (Eds.), Advances in Information Retrieval, Springer Nature Switzerland, Cham, 2025, pp. 467–478.
- [5] F. Alam, J. M. Struß, T. Chakraborty, S. Dietze, S. Hafid, K. Korre, A. Muti, P. Nakov, F. Ruggeri, S. Schellhammer, V. Setty, M. Sundriyal, K. Todorov, V. Venktesh, Overview of the CLEF-2025 CheckThat! Lab: Subjectivity, fact-checking, claim normalization, and retrieval, in: J. Carrillo-de Albornoz, J. Gonzalo, L. Plaza, A. García Seco de Herrera, J. Mothe, F. Piroi, P. Rosso, D. Spina, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Sixteenth International Conference of the CLEF Association (CLEF 2025), 2025.
- [6] P. Nakov, D. Corney, M. Hasanain, F. Alam, T. Elsayed, A. Barrón-Cedeño, P. Papotti, S. Shaar, G. Da San Martino, Automated Fact-Checking for Assisting Human Fact-Checkers, in: Z.-H. Zhou (Ed.), Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, International Joint Conferences on Artificial Intelligence Organization, 2021, pp. 4551–4558. URL: <https://doi.org/10.24963/ijcai.2021/619>, survey Track.
- [7] Y.-C. Chang, C. Kruengkrai, J. Yamagishi, XFEVER: Exploring Fact Verification across Languages, in: J.-L. Wu, M.-H. Su (Eds.), Proceedings of the 35th Conference on Computational Linguistics and Speech Processing (ROCLING 2023), The Association for Computational Linguistics and Chinese Language Processing (ACLCLP), Taipei City, Taiwan, 2023, pp. 1–11. URL: <https://aclanthology.org/2023.rocling-1.1/>.
- [8] J. Thorne, A. Vlachos, C. Christodoulopoulos, A. Mittal, FEVER: a Large-scale Dataset for Fact Extraction and VERification, in: M. Walker, H. Ji, A. Stent (Eds.), Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 809–819. URL: <https://aclanthology.org/N18-1074/>. doi:10.18653/v1/N18-1074.
- [9] A. Gupta, V. Srikumar, X-Fact: A New Benchmark Dataset for Multilingual Fact Checking, in: C. Zong, F. Xia, W. Li, R. Navigli (Eds.), Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), Association for Computational Linguistics, Online, 2021, pp. 675–682. URL: <https://aclanthology.org/2021.acl-short.86/>. doi:10.18653/v1/2021.acl-short.86.
- [10] M. Pikuliak, I. Srba, R. Moro, T. Hromadka, T. Smoleň, M. Melišek, I. Vykopal, J. Simko, J. Podroužek, M. Bielikova, Multilingual Previously Fact-Checked Claim Retrieval, in: H. Bouamor, J. Pino, K. Bali (Eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2023, pp. 16477–16500. URL: <https://aclanthology.org/2023.emnlp-main.1027/>. doi:10.18653/v1/2023.emnlp-main.1027.
- [11] A. Singhal, V. Shao, G. Sun, R. Ding, J. Lu, K. Zhu, A Comparative Study of Translation Bias and Accuracy in Multilingual Large Language Models for Cross-Language Claim Verification, 2024. URL: <https://arxiv.org/abs/2410.10303>. arXiv: 2410.10303.
- [12] R. F. Cekinel, P. Karagoz, Ç. Cöltekin, Cross-Lingual Learning vs. Low-Resource Fine-Tuning: A Case Study with Fact-Checking in Turkish, in: N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti, N. Xue (Eds.), Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), ELRA and ICCL, Torino, Italia, 2024, pp. 4127–4142. URL: <https://aclanthology.org/2024.lrec-main.368/>.
- [13] R. Panchendarajan, A. Zubiaga, Entity-aware Cross-lingual Claim Detection for Automated Fact-checking, 2025. URL: <https://arxiv.org/abs/2503.15220>. arXiv: 2503.15220.
- [14] A. Singhal, T. Law, C. Kassner, A. Gupta, E. Duan, A. Damle, R. L. Li, Multilingual Fact-Checking

- using LLMs, in: D. Dementieva, O. Ignat, Z. Jin, R. Mihalcea, G. Piatti, J. Tetreault, S. Wilson, J. Zhao (Eds.), Proceedings of the Third Workshop on NLP for Positive Impact, Association for Computational Linguistics, Miami, Florida, USA, 2024, pp. 13–31. URL: <https://aclanthology.org/2024.nlp4pi-1.2/>. doi:10.18653/v1/2024.nlp4pi-1.2.
- [15] I. Vykopal, M. Pikuliak, S. Ostermann, M. Šimko, Generative Large Language Models in Automated Fact-Checking: A Survey, 2024. URL: <https://arxiv.org/abs/2407.02351>. arXiv: 2407.02351.
 - [16] A. Kazemi, K. Garimella, D. Gaffney, S. A. Hale, Claim Matching Beyond English to Scale Global Fact-Checking, in: C. Zong, F. Xia, W. Li, R. Navigli (Eds.), 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), Association for Computational Linguistics, Online, 2021, pp. 4504–4517. URL: <https://aclanthology.org/2021.acl-long.347/>. doi:10.18653/v1/2021.acl-long.347.
 - [17] A. Hrckova, R. Moro, I. Srba, J. Simko, M. Bielikova, Autonomation, not Automation: Activities and Needs of Fact-checkers as a Basis for Designing Human-Centered AI Systems, 2024. URL: <https://arxiv.org/abs/2211.12143>. arXiv: 2211.12143.
 - [18] M. Sundriyal, T. Chakraborty, P. Nakov, From chaos to clarity: Claim normalization to empower fact-checking, in: H. Bouamor, J. Pino, K. Bali (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2023, Association for Computational Linguistics, Singapore, 2023, pp. 6594–6609. URL: <https://aclanthology.org/2023.findings-emnlp.439/>. doi:10.18653/v1/2023.findings-emnlp.439.
 - [19] S. Shaar, F. Haouari, W. Mansour, M. Hasanain, N. Babulkov, F. Alam, G. Da San Martino, T. Elsayed, P. Nakov, Overview of the CLEF-2021 CheckThat! lab task 2 on detecting previously fact-checked claims in tweets and political debates (2021).
 - [20] S. Shaar, N. Babulkov, G. Da San Martino, P. Nakov, That is a Known Lie: Detecting Previously Fact-Checked Claims, in: D. Jurafsky, J. Chai, N. Schluter, J. Tetreault (Eds.), Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 3607–3618. URL: <https://aclanthology.org/2020.acl-main.332/>. doi:10.18653/v1/2020.acl-main.332.
 - [21] I. Larraz, R. Míguez, F. Sallicati, Semantic similarity models for automated fact-checking: Claim-Check as a claim matching tool, Profesional de la Información 32 (2023).
 - [22] S. Shaar, F. Alam, G. Da San Martino, P. Nakov, The Role of Context in Detecting Previously Fact-Checked Claims, in: M. Carpuat, M.-C. de Marneffe, I. V. Meza Ruiz (Eds.), Findings of the Association for Computational Linguistics: NAACL 2022, Association for Computational Linguistics, Seattle, United States, 2022, pp. 1619–1631. URL: <https://aclanthology.org/2022.findings-naacl.122/>. doi:10.18653/v1/2022.findings-naacl.122.
 - [23] J. Ni, M. Shi, D. Stammbach, M. Sachan, E. Ash, M. Leippold, AFaCTA: Assisting the Annotation of Factual Claim Detection with Reliable LLM Annotators, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 1890–1912. URL: <https://aclanthology.org/2024.acl-long.104/>. doi:10.18653/v1/2024.acl-long.104.
 - [24] D. Metropolitansky, J. Larson, Towards Effective Extraction and Evaluation of Factual Claims, 2025. URL: <https://arxiv.org/abs/2502.10855>. arXiv: 2502.10855.
 - [25] N. Reimers, I. Gurevych, Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2019. URL: <https://arxiv.org/abs/1908.10084>.
 - [26] A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jégou, T. Mikolov, FastText.zip: Compressing text classification models, arXiv preprint arXiv:1612.03651 (2016).
 - [27] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, A. Yang, A. Fan, A. Goyal, A. Hartshorn, A. Yang, A. Mitra, A. Sravankumar, A. Korenev, A. Hinsvark, et al., The Llama 3 Herd of Models, 2024. URL: <https://arxiv.org/abs/2407.21783>. arXiv: 2407.21783.

- [28] A. Bercovich, I. Levy, I. Golan, M. Dabbah, R. El-Yaniv, O. Puny, I. Galil, Z. Moshe, T. Ronen, N. Nabwani, et al., Llama-Nemotron: Efficient Reasoning Models, 2025. URL: <https://arxiv.org/abs/2505.00949>. arXiv: 2505.00949.
- [29] A. Yang, B. Yang, B. Hui, B. Zheng, B. Yu, C. Zhou, C. Li, C. Li, D. Liu, F. Huang, et al., Qwen2 Technical Report, 2024. URL: <https://arxiv.org/abs/2407.10671>. arXiv: 2407.10671.
- [30] A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv, et al., Qwen3 Technical Report, 2025. URL: <https://arxiv.org/abs/2505.09388>. arXiv: 2505.09388.
- [31] G. Team, A. Kamath, J. Ferret, S. Pathak, N. Vieillard, R. Merhej, S. Perrin, T. Matejovicova, A. Ramé, M. Rivière, et al., Gemma 3 Technical Report, 2025. URL: <https://arxiv.org/abs/2503.19786>. arXiv: 2503.19786.
- [32] K. Ociepa, Łukasz Flis, K. Wróbel, A. Gwoździej, R. Kinas, Bielik 11b v2 technical report, 2025. URL: <https://arxiv.org/abs/2505.02410>. arXiv: 2505.02410.
- [33] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, LoRA: Low-Rank Adaptation of Large Language Models, in: The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022, OpenReview.net, 2022. URL: <https://openreview.net/forum?id=nZeVKeeFYf9>.
- [34] X. Zhang, Y. Zhang, D. Long, W. Xie, Z. Dai, J. Tang, H. Lin, B. Yang, P. Xie, F. Huang, M. Zhang, W. Li, M. Zhang, mGTE: Generalized Long-Context Text Representation and Reranking Models for Multilingual Text Retrieval, 2024. URL: <https://arxiv.org/abs/2407.19669>. arXiv: 2407.19669.
- [35] A. Üstün, V. Aryabumi, Z. Yong, W.-Y. Ko, D. D’souza, G. Onilude, N. Bhandari, S. Singh, H.-L. Ooi, A. Kayid, F. Vargus, P. Blunsom, S. Longpre, N. Muennighoff, M. Fadaee, J. Kreutzer, S. Hooker, Aya Model: An Instruction Finetuned Open-Access Multilingual Language Model, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 15894–15939. URL: <https://aclanthology.org/2024.acl-long.845/>. doi:10.18653/v1/2024.acl-long.845.
- [36] Q. Hu, Q. Long, W. Wang, Decomposition dilemmas: Does claim decomposition boost or burden fact-checking performance?, in: L. Chiruzzo, A. Ritter, L. Wang (Eds.), Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), Association for Computational Linguistics, Albuquerque, New Mexico, 2025, pp. 6313–6336. URL: <https://aclanthology.org/2025.nacl-long.320/>. doi:10.18653/v1/2025.nacl-long.320.
- [37] NVIDIA Corporation, NIM Platform, 2023. URL: <https://developer.nvidia.com/nim>.
- [38] Meta AI, Llama-4-Scout-17B-16E, 2025. URL: <https://huggingface.co/meta-llama/Llama-4-Scout-17B-16E>.
- [39] Mistral AI, Mistral-Small-24B-Base-2501, 2025. URL: <https://huggingface.co/mistralai/Mistral-Small-24B-Base-2501>.
- [40] Groq Inc., Quickstart - GroqDocs, <https://console.groq.com/docs/quickstart>, 2025. URL: <https://console.groq.com/docs/quickstart>, accessed: 2025-05-26.

A. Computational Resources

For our experiments, we leveraged a computational infrastructure consisting of A40 PCIe 40GB, H100 NVL 94GB NVIDIA GPUs, while our experiments ran in parallel on multiple GPUs. In addition, the Polish experiments were conducted on a local workstation equipped with an NVIDIA GeForce RTX 3080 GPU and utilising the NVIDIA NIM platform [37].

B. Details on Parameter-Efficient Fine-Tuning

The adapters were tuned for each language separately, using the filtered training data. In the pilot experiments with German we found that the maximum sequence length 2048, learning rate 2e-4, and



Figure 3: Zero-shot prompt for Direct and Summarization-based normalization based experiments for the *monolingual setting*.

linear scheduler work well for the normalized claim generation, thus we re-used these hyperparameters for training adapters in all languages. We use $r = 32$ and $lora_alpha = 32$ with $lora_dropout = 0$, and train the adapters for 3 epochs to avoid overfitting. At inference time we set max_new_tokens to 256, and generate the claims with the following hyperparameters: $temperature = 0.7$, $top_p = 0.8$, $top_k = 20$.

C. Prompting Experiments

C.1. Prompt Templates

In this section, we present the system and prompt templates used for specific prompting experiments. Figure 4 and Figure 5 illustrate the prompt templates for the Direct Normalization and Summarization-Based Normalization approaches, respectively. Each template includes two demonstration examples. The prompt design emphasizes key aspects such as maintaining focus on important points, eliminating redundancy, ensuring objectivity in claims, and using clear, simple language. In the zero-shot approach for *monolingual experiments*, we assign a fact-checker role to LLMs and prompt it to generate a normalized claim from an unstructured input post, see Figure 3 for the prompt template. For the *zero-shot experiments* we use the direct normalization prompt without any demonstrations from train set.

For the zero-shot and few-shot prompting experiments, described in Section 3.2.3, we used the system prompt shown in Figure 6. Our zero-shot prompt is shown in Figure 7, while the extended version for the few-shot prompting is illustrated in Figure 8. In few-shot prompting, we replace `{examples}` with a list of social media posts along with the normalized claims. The number of demonstrations depends on the setting and whether we used 1, 2, 5 or 10-shot prompting.

Figure 10 and Figure 11 show, respectively, the system prompt and user prompt for few-shot prompting experiments for the Polish language.

C.2. Post Overlap in Development Data

Figure 12 presents the overlap between the gold training and developing data.

C.3. Additional Results

Few-Shot Prompting. Table 8 presents the results for varying numbers of demonstrations for the few-shot prompting. In this scenario, we employed instructions written in the *English* language. Overall, **both Qwen3 models consistently outperformed the Gemma3 model** using few-shot prompting. The best averaged performance was achieved by Qwen3 32B with 10-shot using unfiltered data. This demonstrated that Qwen3 are better equipped to handle the demonstrations and they also show stronger multilingual capabilities.

Increasing the number of demonstrations in the prompt generally improves performance, particularly for large models. For example Qwen3 32B improved from 0.315 1-shot, unfiltered) to 0.375 (10-shot, unfiltered). Moreover, using unfiltered data often led to better results on average.

Similarly to the results using zero-shot and 10-shot prompting, **Latin-script Indo-European languages yielded the highest scores**, reflecting both their prevalence in pre-training data and linguistic

Create a best normalized claim from the unstructured data.

Follow these guidelines:

1. Focus on the main message — Extract only the most important factual statement from the post.
2. Remove redundancy — Ignore repetition, extraneous details, and any irrelevant content (hashtags, usernames, etc.).
3. Keep it objective — Avoid opinions, judgments, or speculation.
4. Use simple language— Rephrase complex or convoluted sentences into clear, direct statements.
5. Formatting — Use ONLY this format: Normalized Claim: [your claim here]

Example 1:

Post: 'Lieutenant Retired General Asif Mumtaz appointed as Chairman Pakistan Medical Commission PMC Lieutenant Retired General Asif Mumtaz appointed as Chairman Pakistan Medical Commission PMC Lieutenant Retired General Asif Mumtaz appointed as Chairman Pakistan Medical Commission PMC None.'

Normalized Claim: 'Pakistani government appoints former army general to head medical regulatory body.'

Example 2:

Post: A priceless clip of 1970 of Bruce Lee playing Table Tennis with his Nan-chak !! His focus on speed A priceless clip of 1970 of Bruce Lee playing Table Tennis with his Nan-chak !! His focus on speed A priceless clip of 1970 of Bruce Lee playing Table Tennis with his Nan-chak !! His focus on speed None

Normalized Claim: Late actor and martial artist Bruce Lee playing table tennis with a set of nunchucks.

Now process this claim:

{post}

Figure 4: Prompt template for the Direct Normalization Approach.

Create a summary from the unstructured data in the form of a normalized claim.

Follow these guidelines:

1. Focus on the main message — Extract only the most important factual statement from the post.
2. Remove redundancy— Ignore repetition, extraneous details, and any irrelevant content (hashtags, usernames, etc.).
3. Keep it objective — Avoid opinions, judgments, or speculation.
4. Use simple language— Rephrase complex or convoluted sentences into clear, direct statements.
5. Formatting — Use ONLY this format: Normalized Claim: [your claim here]

Example 1:

Post: 'Lieutenant Retired General Asif Mumtaz appointed as Chairman Pakistan Medical Commission PMC Lieutenant Retired General Asif Mumtaz appointed as Chairman Pakistan Medical Commission PMC Lieutenant Retired General Asif Mumtaz appointed as Chairman Pakistan Medical Commission PMC None.'

Normalized Claim: 'Pakistani government appoints former army general to head medical regulatory body.'

Example 2:

Post: A priceless clip of 1970 of Bruce Lee playing Table Tennis with his Nan-chak !! His focus on speed A priceless clip of 1970 of Bruce Lee playing Table Tennis with his Nan-chak !! His focus on speed A priceless clip of 1970 of Bruce Lee playing Table Tennis with his Nan-chak !! His focus on speed None

Normalized Claim: Late actor and martial artist Bruce Lee playing table tennis with a set of nunchucks.

Now process this claim:

{post}

Figure 5: Prompt template for the Summarization-Based Normalization.

You are an expert in misinformation detection and fact-checking. Your task is to identify the central claim in the given post while preserving its original language.

Figure 6: System prompt used for zero-shot and few-shot prompting experiments.

similarity to *English*. In contrast, languages using non-Latin scripts showed lower performance, highlighting the challenges in multilingual generalization for underrepresented scripts. However, there are some exceptions, such as *Indonesian* and *Tamil*, where the best performance was over 0.42.

Direct and Summarization-Based Normalization Methods. As additional experiments for the *monolingual setting*, we experimented with three different approaches: two few-shot prompting methods and zero-shot prompting. In the first approach (hereafter referred to as *Direct Normalization Approach*),

You are an expert in misinformation detection and fact-checking. Your task is to identify the central claim in the given post while preserving its original language.

The central claim should meet the following criteria:

- **Verifiable**: It must be a factual assertion that can be checked against evidence.
- **Concise**: It should be a single, clear sentence that captures the main claim of the post.
- **Socially impactful**: It should be a statement that could influence public opinion, health, or policy.
- **Free from rhetorical elements**: Do not include opinions, rhetorical questions, or unnecessary context.
- **Preserve Original Language**: The output should be in the same language as the input post.

Output only the central claim without additional explanation or formatting.

Post: {post}

Normalized claim:

Figure 7: Zero-Shot prompt used for *monolingual* and *zero-shot* settings across 5 LLMs.

You are an expert in misinformation detection and fact-checking. Your task is to identify the central claim in the given post while preserving its original language.

The central claim should meet the following criteria:

- **Verifiable**: It must be a factual assertion that can be checked against evidence.
- **Concise**: It should be a single, clear sentence that captures the main claim of the post.
- **Socially impactful**: It should be a statement that could influence public opinion, health, or policy.
- **Free from rhetorical elements**: Do not include opinions, rhetorical questions, or unnecessary context.
- **Preserve Original Language**: The output should be in the same language as the input post.

Output only the central claim without additional explanation or formatting.

Examples: {examples}

Post: {post}

Normalized claim:

Figure 8: Few-Shot prompt used for *monolingual* and *zero-shot* settings across 5 LLMs.

we instructed LLMs to generate the most accurate normalized claims directly from unstructured data. The prompt template used for *Direct Normalization* is illustrated in Figure 4. In the second method (*Summarization-Based Normalization*), we summarized the unstructured data into a normalized claim, as shown in the prompt template in Figure 5. For both approaches, we included two demonstrations that were randomly selected from the training set as references and evaluated the performance on the development set. In the zero-shot approach for the *monolingual experiments*, where the LLMs rely solely on their pre-trained knowledge, we provided instructions without including any training examples. For the *monolingual experiments* for each language, we used the instruction in the specific language by translating the prompt into that language using the Google Translate API.

In the *zero-shot setting* with 7 languages, we relied on the direct normalization approach, as it produced the best results in the *monolingual experiments*. In this case, however, we instructed LLMs using prompts written entirely in English, without any translated prompts or demonstrations. This setup evaluates the model’s ability to generalize across languages using its pre-trained multilingual capabilities.

For these experiments, we selected three LLMs, specifically Llama4 Scout [38], Llama3.3 Instruct 70B [27] and Mistral Saba [39]. Additionally, for running the experiments, we used the Groq API [40], configured with a maximum output limit of 80 tokens and a temperature setting of 0.3.

Table 9 presents the results for Mistral Saba, Llama 3.3 Instruct, and Llama 4 Scout in the *monolingual setting* on the development set, using zero-shot, direct and summarization-based normalization approaches. Among these, the direct normalization approach with Mistral Saba

Post: {post}

Krok 1: Przeanalizuj treść tekstu i zidentyfikuj kluczowe informacje. Krok 2: Określ główny wątek lub temat tekstu. Krok 3: Zidentyfikuj najważniejsze słowa i frazy w tekście. Krok 4: Określ relacje między kluczowymi informacjami. Krok 5: Zidentyfikuj główny problem lub wyzwanie opisane w tekście. Krok 6: Określ, kto lub co jest głównym podmiotem tekstu. Krok 7: Zidentyfikuj najważniejsze skutki lub konsekwencje opisane w tekście. Krok 8: Określ, jaki jest główny cel lub zamierzenie tekstu. Krok 9: Zidentyfikuj najważniejsze słowa i frazy, które mogą być użyte w twierdzeniu znormalizowanym. Krok 10: Stwórz twierdzenie znormalizowane, które podsumowuje treść tekstu w sposób zwięzły i precyzyjny.

Odpowiedź: (twierdzenie znormalizowane, nie dłuższe niż 9 wyrazów)

Wyświetl tylko odpowiedź !!!

Nie wyświetlaj żadnych komentarzy ani uwag !!!

English Translation:

Post: {post}

Step 1: Analyse the text content and identify key information. Step 2: Determine the main thread or topic of the text. Step 3: Identify the most important words and phrases in the text. Step 4: Determine the relationships between key information. Step 5: Identify the main problem or challenge described in the text. Step 6: Determine who or what is the main subject of the text. Step 7: Identify the most important effects or consequences described in the text. Step 8: Determine what is the main goal or intention of the text. Step 9: Identify the most important words and phrases that can be used in a normalized statement. Step 10: Create a normalized statement that summarizes the text content in a concise and precise manner.

Answer: (normalized statement, no longer than 9 words)

Display only the answer !!!

Do not display any comments or hints !!!

Figure 9: Polish-CoT, original prompt in Polish and translation into English.

Your task is to simplify a noisy, unstructured social media post into a concise form while preserving the core assertion. You will be given a post and you need to generate a normalized claim. Please respond with the normalized claim.

The normalised claim must contain a maximum of 10 words or fewer. The normalised claim must be in the Polish language only.

Figure 10: System prompt for few-shot prompting experiments for Polish. The model was asked to limit the answer up to 10 words, as based on the statistic analysis the average normalized claim for Polish dataset contains about 10 words.

Shot 1: Post: Example 1
 Normalized Claim: Example 1
 Shot 2: Post: Example 2
 Normalized Claim: Example 2
 Shot 3: Post: Example 3
 Normalized Claim: Example 3

Your Task: Given a noisy, unstructured social media post, simplify it into a concise form while preserving the core assertion.
 Please respond with the normalized claim for the following post: {post}
 do not display any comments

Figure 11: User prompt for few-shot prompting experiments for Polish.

achieves the highest average score on the development set. The lowest average score is observed with **Mistral Saba** using the zero-shot approach. The difference between the highest and lowest average score is 0.083. We observe that, all three models perform better than zero-shot setting with direct and summarization-based normalization.

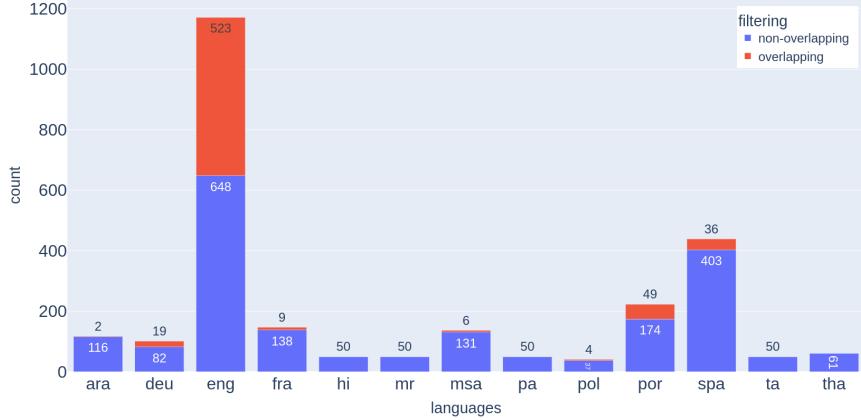


Figure 12: Post overlap between the gold train and development data.

Table 8

LLM performance in the *monolingual setting* on the development set using various numbers of demonstrations within the prompt. The *Fil.* column specifies the few-shot prompting setup: \checkmark denotes that filtered data was used to sample demonstrations, whereas an empty cell indicates the use of unfiltered data. Best results for each language are in **bold** and the second-best are underlined.

Model	# of Shots	Fil.	ara	deu	eng	fra	hi	mr	msa	pa	pol	por	spa	ta	tha	Avg.
Qwen3 (8B)	1-shot		0.367	0.253	0.491	0.326	0.235	0.283	0.348	0.312	0.209	0.423	0.410	0.413	0.161	0.325
	1-shot	\checkmark	<u>0.366</u>	0.232	0.373	0.331	0.229	0.319	0.348	0.312	0.198	0.425	0.407	0.424	0.161	0.317
	2-shot		0.341	0.237	0.551	0.340	0.255	0.302	0.344	0.311	0.234	0.435	0.415	0.430	0.156	0.335
	2-shot	\checkmark	0.333	0.209	0.405	0.339	0.254	0.325	0.341	0.324	0.216	0.458	0.425	0.393	0.185	0.324
	5-shot		0.340	0.230	0.563	0.345	0.247	0.353	0.351	0.329	0.222	0.473	0.425	0.397	0.184	0.343
	5-shot	\checkmark	0.345	0.259	0.420	0.351	<u>0.266</u>	0.307	0.336	0.319	0.220	0.462	0.437	0.388	0.204	0.332
	10-shot		0.355	0.241	0.550	0.373	0.234	0.309	0.361	0.341	0.237	0.460	0.428	0.390	0.192	0.344
	10-shot	\checkmark	0.342	0.244	0.432	0.375	0.264	0.333	0.345	0.331	0.224	0.463	0.445	0.412	0.175	0.337
	1-shot		0.261	0.203	0.318	0.303	0.224	0.274	0.265	0.185	0.213	0.368	0.331	0.341	0.164	0.266
	1-shot	\checkmark	0.272	0.220	0.285	0.304	0.209	0.288	0.264	0.182	0.221	0.373	0.330	0.341	0.142	0.264
Gemma3 (27B)	2-shot		0.282	0.229	0.397	0.305	0.217	0.283	0.296	0.261	0.232	0.379	0.364	0.384	0.187	0.294
	2-shot	\checkmark	0.268	0.227	0.322	0.326	0.234	0.303	0.280	0.226	0.231	0.412	0.354	0.389	0.124	0.284
	5-shot		0.288	0.241	0.460	0.348	0.227	0.312	0.341	0.266	0.283	0.442	0.396	0.327	0.197	0.318
	5-shot	\checkmark	0.303	0.252	0.340	0.350	0.217	0.308	0.352	0.345	0.237	0.463	0.381	0.340	0.207	0.315
	10-shot		0.312	<u>0.280</u>	0.479	0.371	0.231	0.308	<u>0.404</u>	0.310	0.236	0.472	0.428	0.349	0.222	0.339
	10-shot	\checkmark	0.306	0.261	0.357	0.368	0.226	0.327	0.364	0.314	0.241	0.468	0.413	0.348	0.196	0.322
	1-shot		0.301	0.241	0.531	0.377	0.216	0.307	0.321	0.275	0.229	0.426	0.412	0.339	0.116	0.315
Qwen3 (32B)	1-shot	\checkmark	0.302	0.263	0.402	0.384	0.225	0.311	0.321	0.274	0.222	0.437	0.407	0.338	0.122	0.308
	2-shot		0.344	0.255	0.541	0.366	0.236	0.325	0.335	0.284	0.248	0.470	0.426	0.370	0.141	0.334
	2-shot	\checkmark	0.346	0.250	0.412	0.357	0.240	0.335	0.330	0.284	0.231	0.483	0.424	0.409	0.125	0.325
	5-shot		0.358	0.267	<u>0.570</u>	0.379	0.257	<u>0.349</u>	0.387	0.311	<u>0.267</u>	0.515	0.454	0.410	0.190	<u>0.363</u>
	5-shot	\checkmark	0.349	<u>0.280</u>	0.423	0.381	0.239	0.336	0.360	<u>0.345</u>	0.230	0.511	0.463	0.350	0.173	0.341
	10-shot		0.348	0.271	0.587	<u>0.409</u>	0.293	0.324	0.420	<u>0.350</u>	0.224	<u>0.536</u>	0.485	0.416	<u>0.211</u>	0.375
	10-shot	\checkmark	0.367	0.303	0.432	0.413	0.258	0.334	0.403	0.319	0.265	<u>0.537</u>	<u>0.474</u>	0.342	0.152	0.354

Table 9

Monolingual results on development set across three models (Mistral Saba, Llama3.3 Instruct, and Llama4 Scout), with average score across 13 languages.

Model	Approach	ara	deu	eng	fra	hi	mr	msa	pa	pol	por	spa	ta	tha	Avg.
Mistral Saba	Zero-shot	0.253	0.115	0.199	0.228	0.135	0.108	0.166	0.188	0.141	0.225	0.207	0.257	0.113	0.179
	Direct _{Nor}	0.347	0.210	0.293	0.295	0.228	0.152	0.258	0.294	0.269	0.290	0.294	0.321	0.160	0.262
	Summarization _{Nor}	0.341	0.218	0.298	0.289	0.215	0.157	0.252	0.298	0.253	0.296	0.295	0.339	0.139	0.260
Llama3.3 Instruct	Zero-shot	0.284	0.166	0.238	0.259	0.175	0.116	0.210	0.135	0.180	0.264	0.238	0.112	0.109	0.191
	Direct _{Nor}	0.333	0.229	0.286	0.322	0.253	0.172	0.281	0.183	0.246	0.328	0.304	0.172	0.136	0.250
	Summarization _{Nor}	0.341	0.242	0.289	0.324	0.252	0.182	0.273	0.183	0.250	0.327	0.307	0.157	0.154	0.252
Llama4 Scout	Zero-shot	0.132	0.121	0.199	0.258	0.171	0.119	0.184	0.154	0.165	0.243	0.254	0.260	0.150	0.199
	Direct _{Nor}	0.350	0.230	0.283	0.296	0.250	0.143	0.286	0.228	0.266	0.286	0.299	0.246	0.190	0.257
	Summarization _{Nor}	0.341	0.218	0.270	0.289	0.265	0.132	0.263	0.223	0.275	0.309	0.310	0.248	0.173	0.255