Document-level Claim Extraction and Decontextualization for FactChecking

Project Report

xpolic05, xmakai00, xklein19

May 18, 2025

Abstract

We address the task of decomposing Czech texts into checkworthy atomic claims to support automatic fact-checking. Our work builds on the English-based LOKI system, which we adapt for Czech by leveraging machine translation and evaluating its performance on Czech inputs. We experiment with few-shot prompting strategy and Chain of Thought reasoning to improve atomic claim extraction and compare their effectiveness. Our results show that, with proper translation, existing models can produce useful decompositions of Czech texts.

Keywords: fact-checking, atomic claims, claim decomposition, multilingual NLP.

Contents

1 Task Definition

This project aims to improve systems for automatic fact-checking, specifically targeting the task of text decomposition through the extraction of so-called atomic claims.

Automatic fact-checking generally involves three stages: claim extraction, evidence retrieval, and claim verification. Our work focuses on the first stage, where atomic claims—statements that cannot be further divided—are extracted. These simplify the downstream tasks and must be factual, verifiable, and ideally practically useful.

We target Czech-language texts and evaluate various models and techniques, including few-shot, and Chain-of-Thought (CoT) prompting.

2 Review of Existing Solutions and Relevant Information

Current solutions for atomic claim extraction are predominantly designed for English. Furthermore, they usually already work with atomic, often artificial claims. Only a small number of solution adress also natural document-level decomposition.

One of the most promising tools we found is **Loki** [2], an open-source system designed to automate fact-checking. It offers a complete pipeline that includes the decomposition of long texts into individual claims, assessment of checkworthiness, query generation for evidence retrieval, web crawling, and claim verification.

While Loki was developed for English and Chinese, we found it performs surprisingly well on Czech texts.

3 Description of Our Solution

We adapted Loki, focusing only on atomic claim generation and checkworthiness evaluation. We extended Loki's LLM client to support local models via Ollama, enabling testing beyond OpenAI/Anthropic APIs supported by original tool. This allowed us to test a wider variety of models and evaluate which perform best on our task.

As there are no Czech-specific datasets for this task, we created them ourselves.

- Translated version of Factcheck-Bench [**?**], one of the few English datasets with claim-source pairs.
- Manually annotated dataset of Czech comments from internet articles.

We processed both datasets using a custom Python script (decompose.py) allowing model and prompt selection.

Models tested include phi-4 (quantized and full-precision) and Qwen 2.5, chosen for their strong performance on Czech tasks, as reported in **BenCzechMark** paper [1]. We evaluated Czech vs. English prompts, Chain-of-Thought reasoning, and few-shot learning.

3.1 Datasets

We manually constructed a dataset of atomic claims derived from 60 user comments on Czech internet news articles. To complement this, we also incorporated translated pairs of source texts and atomic claims from the Factcheck-Bench dataset. Compared to our manually curated dataset, the translated dataset has on average a higher number of claims per source text and longer individual claims.

These datasets offer complementary characteristics, supporting robust evaluation.

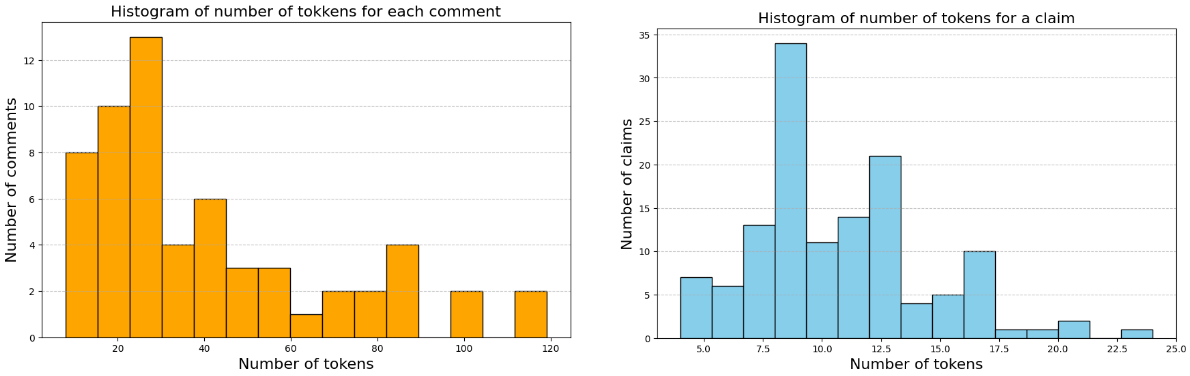

Figure 1: The graphs below present histograms illustrating the distribution of token counts for both the source texts (comments) and the atomic claims in our manually created dataset. On average, each comment contains approximately three atomic claims.

3.2 Claim decomposition tool

3.2.1 LLM client

We implemented our custom client based on BaseClient to enable the use of local models via Ollama.

3.2.2 Decomposition

The modified Loki tools are utilized via the decompose.py script, which manages the loading of the specified large language model (LLM) and processes all comments and source texts from the dataset. The script generates and saves the resulting outputs. Additionally, it supports the use of customized prompts during inference, allowing for greater flexibility in model behavior.

3.3 Evaluation

To assess the quality of the generated atomic claims, we employed two complementary evaluation metrics: ROUGE-L and edit distance.

3.3.1 ROUGE-L

We compute the ROUGE-L score to assess the sequential similarity between generated and reference claims. ROUGE-L measures the length of the longest common subsequence (LCS), capturing not only content overlap but also the order of words. Higher ROUGE-L scores indicate closer alignment to the reference in both content and structure.

3.3.2 Edit Distance

Edit distance measures the minimum number of operations (insertions, deletions, or substitutions) required to transform a generated claim into the corresponding reference claim. This metric captures structural differences and is particularly useful for identifying paraphrasing or rewording that may not be reflected in n-gram overlap scores.

4 Experiments

4.1 Setup

- **Metrics used:** ROUGE-L and normalized edit distance (scaled 0–1 for comparability).

- **Scoring method:** Scores are averaged across matched atomic claim pairs per source text, using the Hungarian algorithm for optimal one-to-one matching.

- **Evaluation modes:**

  - *Avg*: Considers only successfully matched pairs; ignores extra or missing claims.
  - *Avg_Strict*: Uses padding to penalize over- or under-generation.

- **Good matches:** Pairs with similarity scores above 0.7 are considered good; the total number is reported.

- **Baseline comparison:** We compare our results with the original Loki tool using Claude, a much larger LLM with billions of parameters.

4.2 Model Comparison

We began with the Phi-4 model, as the BenCzechMark benchmark shows it performs surprisingly well on Czech tasks given its size. We also compared the quantized and full-precision versions to assess performance differences.

Additionally, we evaluated Qwen 2.5, a slightly larger model also known for its strong size-to-performance ratio on Czech tasks.

Finally, both datasets were evaluated using Claude Sonnet via the Anthropic API. However, Claude is significantly larger and cannot be run locally, making it more suitable as a high-end baseline rather than our final solution.

4.3 Impact of Translation Strategy for Czech Language

We compared two main approaches for Czech:

1. **Direct Czech Generation with Czech Prompt:** For some models (e.g., certain Phi-4 configurations), the input prompt given to the LLM was formulated in Czech, with the expectation that the model would generate claims and related outputs directly in Czech.

2. **English Generation Followed by Machine Translation:** For other models or baseline configurations (e.g., Qwen 2.5, Claude, or base Phi-4 versions), outputs such as claims and checkworthiness reasons were initially generated in English. To enable evaluation and use within a Czech context, these English outputs were subsequently translated to Czech. This post-hoc translation process is detailed further below.

The objective was to determine which approach yielded higher quality Czech outputs for downstream fact verification tasks.

Post-hoc Translation of Model Outputs to Czech To prepare English outputs for Czech-based evaluation and to facilitate the comparison of language strategies, a programmatic translation step was implemented. This process selectively translated key fields within the JSON result files from designated model configurations that originally produced English text.

The translation was performed using the MarianMT model ‘Helsinki-NLP/opus-mt-en-cs‘ via the Hugging Face ‘transformers‘ library. A Python script automated this by:

- Loading model-generated claims from specified JSON files.

- For configurations flagged for translation, it targeted the ‘claim‘ text and the core content of the ‘checkworthy_reason‘ for conversion from English to Czech. For the ‘checkworthy_-reason‘, text within parentheses was prioritized for translation.

- During this process, claims were also filtered based on their ‘checkworthy‘ status (determined if the ‘checkworthy_reason‘ started with ’Yes’); only checkworthy claims were retained.

- The resulting data, with translated fields and filtered claims, was saved to new JSON files with UTF-8 encoding.

This automated step ensured consistency in converting English outputs to Czech. Figure 4.3 illustrates an example of this transformation process for an output from a Phi-4 model configuration that generated English, before and after the translation and filtering.

```
{
  "source": "Podle původního dokumentu LLaMa (https://arxiv.org/abs/2012.15730) má ...",
  "claims": [
    {
      "id": 0,
      "claim": "Největší verze LLaMa má 4,5 miliardy parametrů.",
      "checkworthy": true,
      "checkworthy_reason": "This statement contains verifiable factual information ...",
      "origin_text": "Podle původního dokumentu LLaMa (https://arxiv.org/abs/2012.15730) ...",
      "start": 0,
      "end": 131
    },
    {
      "id": 2,
      "claim": "Menší verze modelu mají od 110 milionů do 1,5 miliardy parametrů.",
      "checkworthy": true,
      "checkworthy_reason": "This statement contains verifiable factual information ...",
      "origin_text": "s menšími parametry v rozmezí od 110 milionů do 1,5 miliardy...",
      "start": 179,
      "end": 250
    }
  ]
}
```

Figure 2: Example of Phi-4 model output after translation and filtering

4.4 Effect of Prompting Techniques

We tested:

- 1-shot (original prompt)

- 3-shot prompting

- Chain-of-Thought reasoning (CoT)

5 Results

This section presents the outcomes of our experiments evaluating the fact verification system across different configurations and datasets. The performance is measured using ROUGE-L and EDIT DISTANCE-based metrics. The best result is in bold, and second best in italics.

5.1 DATASET 1 Performance

On the Factcheck-Bench dataset, the quantized version of Phi-4, prompted to reason step-by-step, achieved performance closest to the Claude baseline. Notably, the full-precision Phi-4 model with the translated prompt achieved the highest precision—indicating a higher proportion of high-quality matches among generated claims. However, its lower scores on other metrics suggest it produced fewer claims than expected, though those it did generate were of strong quality. The results are simmilar for both metrics.

Table 1: ROUGE-L Performance on DATASET 1 (Factcheck Bench)

Configuration	**Avg_Strict**	**Recall**	**Precision**	**Avg**	**Good_Matches**
phi4_14b-q8_0	0,3338	0,1414	0,2014	0,4999	93
phi4_14b-q8_0_czprompt.	0,4071	0,2932	*0,35*	0,5965	*180*
phi4_14b-q8_0_czprompt_3shot	0,4345	0,2924	0,3392	0,5954	179
phi4_14b-q8_0_czprompt_CoT	*0,4575*	*0,315*	0,3386	**0,6259**	130
phi4_14b-fp16	0,3343	0,1789	0,2363	0,5163	110
phi4_14b-fp16_czprompt	0,4062	0,2937	**0,383**	0,5866	177
qwen2.5_32b-instruct-q4_K_M	0,3493	0,1868	0,2614	0,5151	108
claude_translated	**0,4633**	**0,3424**	0,3114	*0,5994*	**237**

Table 2: EDIT DISTANCE Performance on DATASET 1 (Factcheck Bench)

Configuration	**Avg_Strict**	**Recall**	**Precision**	**Avg**	**Good_Matches**
phi4_14b-q8_0	0,3366	0,1333	0,1917	0,5139	89
phi4_14b-q8_0_czprompt.	0,3946	0,228	0,2817	0,5879	156
phi4_14b-q8_0_czprompt_3shot	0,4245	0,2595	*0,2989*	0,5876	*165*
phi4_14b-q8_0_czprompt_CoT	*0,4444*	*0,2754*	0,2907	**0,619**	120
phi4_14b-fp16	0,3327	0,1528	0,2037	0,5221	101
phi4_14b-fp16_czprompt	0,3969	0,2465	**0,3228**	0,5801	157
qwen2.5_32b-instruct-q4_K_M	0,3479	0,1696	0,2375	0,5147	101
claude_translated	**0,4915**	**0,3137**	0,2881	*0,5995*	**226**

5.2 DATASET 2 Performance

On our custom dataset, several models outperformed the baseline. Notably, the quantized Phi-4 model achieved the highest scores across both evaluation metrics, surpassing even the modified variants that had shown stronger results on the Factcheck-Bench dataset.

Table 3: ROUGE-L Performance on DATASET 2 (comments)

Configuration	**Avg_Strict**	**Recall**	**Precision**	**Avg**	**Good_Matches**
phi4_14b-q8_0	0,2749	0,1198	0,1203	0,3669	13
phi4_14b-q8_0_czprompt	**0,3425**	**0,2401**	**0,2247**	*0,4898*	25
phi4_14b-q8_0_czprompt_3shot	0,3165	0,2052	0,1417	**0,5071**	**28**
phi4_14b-q8_0_czprompt_CoT	0,2976	*0,2139*	0,14	0,4831	20
phi4_14b-fp16	0,2591	0,0877	0,0708	0,3589	7
phi4_14b-fp16_czprompt	*0,3266*	0,188	*0,1989*	0,4842	*27*
qwen2.5_32b-instruct-q4_K_M	0,3021	0,0801	0,0569	0,405	10
claude_translated	0,2768	0,1574	0,109	0,4473	20

Table 4: EDIT DISTANCE Performance on DATASET 2 (comments)

Configuration	**Avg_Strict**	**Recall**	**Precision**	**Avg**	**Good_Matches**
phi4_14b-q8_0	0,3211	0,1264	0,1236	0,4211	13
phi4_14b-q8_0_czprompt	*0,3527*	0,1801	**0,1808**	0,4985	18
phi4_14b-q8_0_czprompt_3shot	0,3248	**0,1961**	0,1395	**0,5109**	**24**
phi4_14b-q8_0_czprompt_CoT	0,3183	0,1741	0,1196	*0,5022*	16
phi4_14b-fp16	0,3143	0,1418	0,1211	0,4211	13
phi4_14b-fp16_czprompt	0,342	0,1634	*0,1802*	0,4896	23
qwen2.5_32b-instruct-q4_K_M	**0,3608**	*0,1946*	0,1611	0,4699	22
claude_translated	0,3176	0,2217	0,1601	0,4852	*23*

6 Conclusions

Our work demonstrates that atomic claim decomposition for Czech-language texts is both feasible and effective using adapted LLM pipelines. By modifying the Loki tool and extending it with local LLM support through Ollama, we enabled experimentation with various models and prompting techniques.
We show that:

- Local models can match or outperform larger LLMs with appropriate prompting.
- Translation strategy and prompt design significantly affect performance.
- Surprisingly, quantized models can outperform full precision ones in some conditions.

Our experiments also highlight key trade-offs: while some configurations produced fewer claims overall, the claims they did generate were of higher quality. The results suggest that prompt design and translation strategies have a notable impact on performance and should be carefully selected based on the specific downstream application.

References

[1] Martin Fajcik, Martin Docekal, Jan Dolezal, Karel Ondrej, Karel Beneš, Jan Kapsa, Pavel Smrz, Alexander Polok, Michal Hradis, Zuzana Neverilova, Ales Horak, Radoslav Sabol, Michal Stefanik, Adam Jirkovsky, David Adamczyk, Petr Hyner, Jan Hula, and Hynek Kydlicek. Benczechmark : A czech-centric multitask and multimetric benchmark for large language models with duel scoring mechanism, 2024.

[2] Haonan Li, Xudong Han, Hao Wang, Yuxia Wang, Minghan Wang, Rui Xing, Yilin Geng, Zenan Zhai, Preslav Nakov, and Timothy Baldwin. Loki: An open-source tool for fact verification, 2024.