Factiverse and IAI at CheckThat! 2025: Adaptive ICL for Claim Extraction

Notebook for the CheckThat! Lab at CLEF 2025

Pratuat Amatya[1,*], Vinay Setty[2]

[1]Factiverse AS, Prof. Olav Hanssens v. 7 A, 4021 Stavanger, Norway
[2]University of Stavanger, Kjell Arholms gate 41, 4021 Stavanger, Norway

Abstract

In this paper, we describe methods and results of our participation in the 2025 CheckThat! Lab "Task 2: Claim Normalization". The task aims to develop methods to extract concise claims from noisy, unstructured social media posts. The task itself comprised two settings: monolingual and zero-shot setting. In this paper, we focus more on the monolingual setting where training and testing data are available for 13 different languages such as English, German, French, etc. and utilized fine-tuning based methods, zero-shot prompting, and in-context learning (ICL) methods using a fixed and a adaptive number of examples to improve the generations of normalized claims. While the model performance was not consistent across all languages, we achieved notable placements on the organizer's leader-board: fifth-best in English, fourth-best in German, French, Indonesian, and sixth-best in Spanish and Portuguese using fine-tuning based method. Our experiment highlighted the effectiveness of both fine-tuning and ICL based methods with comparable performance scores using labeled test dataset over baseline (zero-shot prompting). The experiment also revealed some challenges associated with the nature of social media data and some limitations associated with models we experimented with, hence providing us insight into how the method could be improved for claim normalization.

Keywords

Claim-normalization, Fact-checking, In-Context Learning, Fine-tuning

1. Introduction

The rapid proliferation of misinformation on social media platforms has created an urgent need for effective automated fact-checking technologies. A foundational step in this pipeline is *claim normalization*—the task of simplifying noisy and unstructured posts into concise, structured statements that can be further verified [1]. The CheckThat! Lab 2025 Task 2 focuses on this crucial preprocessing step, particularly in a multilingual setting, where diverse linguistic and stylistic features pose additional challenges [2].

Claim normalization is inherently challenging due to the variability and informality of social media language, including the use of slang, sarcasm, code-switching, and regional idioms. These characteristics are further exacerbated in multilingual contexts, where linguistic diversity introduces additional complexity. Thus, building robust claim normalization systems that generalize well across languages and domains is of paramount importance.

The 2025 CheckThat! Lab Task 2 provides a rigorous benchmark for this task, encompassing both **monolingual** and **zero-shot** settings over 13 languages. In our submission, we address both tracks but place particular emphasis on the monolingual scenario where annotated training data is available.

We evaluate and compare three core approaches for claim normalization: (i) *zero-shot prompting*, (ii) *fine-tuning large language models (LLM)*, and (iii) *in-context learning (ICL)*. Within the ICL framework, we experiment with two variants: *Fixed In-Context Learning (FICL)*, where a constant number of k examples are included in every prompt, and *Adaptive In-Context Learning (AICL)*, where the number of examples is dynamically adjusted for each input based on semantic similarity. While prior work has

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain
*Corresponding author.
pratuat@factiverse.ai (P. Amatya); vsetty@acm.org (V. Setty)
0009-0004-2593-056X (P. Amatya); 0000-0002-9777-6758 (V. Setty)
© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

investigated fixed-k ICL, our key contribution lies in the implementation and empirical evaluation of the adaptive variant, which leverages cosine similarity in a vector-based retrieval setup to tailor the example set to each input. This enables more context-aware prompting without the need for model retraining or handcrafted selection strategies.

The novelty of our AICL approach lies in its use of a vector-based retrieval mechanism (ChromaDB) combined with a cosine similarity heuristic to select contextually relevant examples from the training data. Unlike traditional ICL methods that rely on a fixed number of examples, AICL tailors the quantity and content of examples based on the semantic proximity of the input, striking a balance between informativeness and cognitive load in the prompt. This adaptation mechanism allows the model to operate more efficiently in heterogeneous data scenarios without requiring explicit model retraining or extensive hyperparameter tuning.

Our results demonstrate that both fine-tuning and ICL-based methods significantly outperform the zero-shot baseline across multiple languages, with AICL matching the performance of FICL while offering superior flexibility and scalability. Specifically, our system achieved top-6 leaderboard placements in several languages, including fifth-best in English and fourth-best in German, French, and Indonesian.

To provide structural overview of this paper, in Section 2, we review related work on claim normalization, emphasizing its significance in the context of fact-checking, as well as an exploration of ICL methods. Section 4 details our experimental setup, including architectural diagrams and formal problem descriptions. The evaluation of four distinct approaches to claim normalization using standard metrics is presented in Section 5, along with a discussion of the results. In Section 6, we outline directions for future research. Finally, Section 7 summarizes our contributions and key findings. We believe these contributions not only advance the state-of-the-art in claim normalization but also provide a reusable framework for multilingual NLP tasks involving few-shot learning and LLM-based generation.

2. Related Work

2.1. Automated Fact-Checking and Claim Detection

Automated fact-checking has become an essential area of research in response to the growing spread of misinformation across digital platforms. The typical fact-checking pipeline consists of multiple stages, including claim detection, evidence retrieval, claim verification, and justification generation. The process begins with claim detection, which identifies statements that require verification.

Early efforts in this field established foundational frameworks for structuring automated fact-checking, encompassing key stages such as claim detection, evidence retrieval, and verdict prediction [3]. These frameworks have since evolved to address challenges in open-domain fact-checking and to incorporate the role of contextual information in verification [4].

An important sub-area of research focuses on identifying check-worthy claims. Early works focused on feature engineering for claim detection [5]. CLEF CheckThat! had several editions of claim detection task [6]. The best solutions have been using fine-tuned transformers and LLM [7]. However, claim extraction is akin to abstractive summarization, which is fundamentally different task than claim detection.

Recent advances have also highlighted the practical implementation of fact-checking systems. Fine-tuned transformer-based models have demonstrated strong performance in multilingual fact-checking scenarios, often outperforming larger language models [8]. Furthermore, integrated tools that combine claim detection and verification within user-friendly interfaces have been developed to support the deployment of automated fact-checking in real-world applications [9].

2.2. Claim Normalization and Summarization

Claim normalization, the process of transforming verbose or unstructured text into concise, check-worthy claims, is crucial for effective fact-checking. This task shares similarities with abstractive

summarization, where the goal is to generate a coherent summary capturing the essence of the source text.

Recently, **C**heck-worthiness **A**ware **C**laim **N**ormalization (**CACN**) framework [1], combining chain-of-thought prompting with claim check-worthiness estimation to decompose complex social media posts into normalized claims, has been introduced by Sundriyal et al. [1]. Their CLAN dataset provides valuable resources for training and evaluating claim normalization systems.

The analogy between claim normalization and summarization lies in their shared objective of distilling essential information. However, claim normalization is more constrained, focusing on extracting factual statements suitable for verification, whereas summarization may include interpretative content.

2.3. In-Context Learning and Adaptive Example Selection

ICL has gained prominence with the advent of LLM, allowing models to perform tasks by conditioning on a few examples provided in the prompt. Traditional ICL methods often use a fixed number of examples (fixed-k ICL), which may not be optimal for all inputs.

Chandra et al. [10] proposed an adaptive approach that predicts the optimal number of in-context examples based on the input, leading to significant improvements in text classification tasks. Their AICL method dynamically adjusts the number of examples, enhancing performance without extensive hyperparameter tuning.

Our work builds upon these insights by implementing an AICL approach for claim normalization. By dynamically selecting the number and content of in-context examples based on semantic similarity thresholds, our method balances the trade-off between providing sufficient contextual information and avoiding prompt overload. This enhances the quality of generated normalized claims, contributing to the development of more robust and scalable fact-checking systems.

3. Dataset

A brief statistics on dataset provided for CheckThat! 2025 claim normalization task (task 2) is presented in Table 1. The dataset is classified into two setups: i) Monolingual, and ii) Zero-shot setup. In monolingual setup, training, development and test dataset are provided for 13 different languages and requires models to be trained, validated and tested in isolation to one particular language, intent being that the model learns language-specific patterns and structures. As for zero-shot setup, only test dataset is provided for 7 different languages. The ambition here is to evaluate generalization capability of a model to unseen languages.

4. Methodology

We conducted a series of experiments on claim normalization using four distinct approaches: (i) Zero-shot prompting (serving as the baseline), (ii) Fine-tuned model approach, (iii) FICL, and (iv) AICL. The task involved using multilingual dataset comprising labeled training, development, and test sets spanning 13 languages. The training set was utilized for model fine-tuning as well as for constructing example sets in the ICL methods, while the development and test sets were reserved for validation and final evaluation. The performance was evaluated using average METEOR score calculated between the generated claims and their corresponding reference normalized claims.

4.1. Zero-shot Prompting

This approach involves using a carefully designed prompt (Figure 6) with explicit instructions to perform claim normalization, without providing any in-context examples.

Table 1
Dataset count by language and setup (Monolingual vs Zero-shot).

Language	Train	Dev	Test
Monolingual Languages			
Arabic (ara)	470	118	100
German (deu)	386	101	100
English (eng)	11374	1171	1285
French (fra)	1174	147	148
Hindi (hi)	1081	50	100
Marathi (mr)	137	50	100
Indonesian (msa)	540	137	100
Pubjabi (pa)	445	50	100
Polish (pol)	163	41	100
Portuguese (por)	1735	223	225
Spanish (spa)	3458	439	439
Tamil (ta)	102	50	100
Thai (tha)	244	61	100
Zero-shot Languages			
Bengali (bn)	–	–	81
Czech (ces)	–	–	123
Greek (ell)	–	–	156
Korean (kor)	–	–	274
Dutch (nld)	–	–	177
Romanian (ron)	–	–	141
Telgu (te)	–	–	116

4.2. Fine-tuning

Building upon the provided baseline, which fine-tunes the mT5 model—a multilingual sequence-to-sequence (Seq2Seq) transformer model—we extended our experimentation by fine-tuning the `google/flan-t5-large` [11] model on the available training data. The `flan-t5-large` model is a variant of T5 [12] that has been instruction-tuned to follow natural language instructions more effectively, which we hypothesized would be advantageous for the check-worthiness estimation task.

However, one key challenge in this task is that not all target languages had sufficient training data available. For languages where training data was provided, the fine-tuned `flan-t5-large` model performed reasonably well, demonstrating its capacity to generalize to this specific task. In contrast, for languages lacking annotated training data, model performance was inherently limited by the absence of in-language supervision.

Although larger variants of the model, such as `google/flan-t5-xl` and `google/flan-t5-xxl`, were considered for experimentation, they could not be fine-tuned in our current setup due to GPU memory constraints and computational resource limitations.

To address the data scarcity issue for low-resource languages, we explored data augmentation through machine translation. In this approach, training data from high-resource languages was translated into the low-resource target languages to create synthetic training data. This method aimed to enhance the model's ability to perform cross-lingual generalization and improve performance in languages for which no direct training data was provided. While this translation-based strategy is not without limitations—such as potential translation artifacts and domain mismatch—it proved to be a promising direction for improving coverage and robustness of our multilingual fact-checking pipeline.

4.3. In-Context Learning (ICL)

A key strength of LLM lies in their ability to leverage examples and instructions embedded within the input prompt. ICL leverages this capability by incorporating task-relevant examples into the prompt (Figure 7) to guide generation. Our ICL strategies were inspired by the work of Chandra et al. [10], who

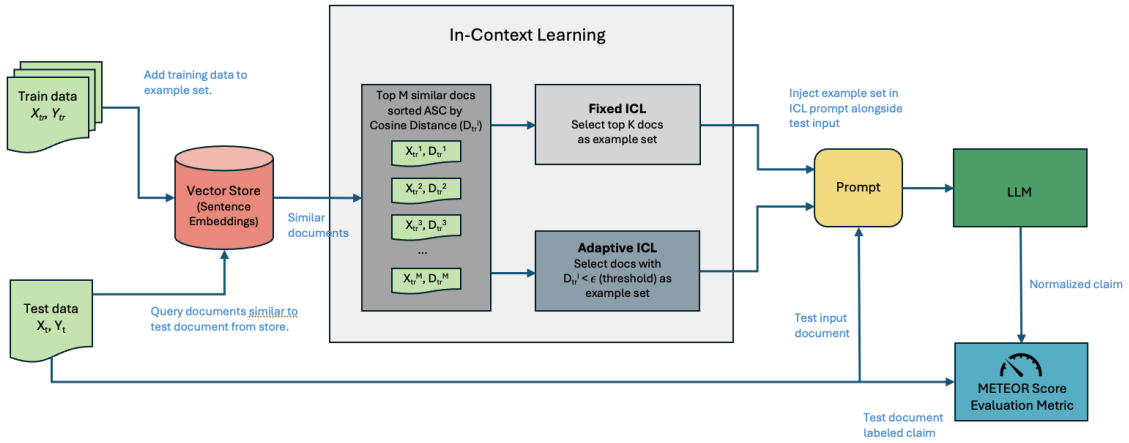

Figure 1: Block diagram for ICL based method for claim normalization. As illustrated inside the grey block, only distinction between FICL and AICL method is on selection of examples set to be injected in claim normalization prompt. Final evaluation metric is average of the METEOR score of all test inputs.

demonstrated that dynamically adjusting the number of in-context examples can substantially improve performance in text classification tasks. Their AICL method showed an improvement of 17% over the fixed-example baseline.

In Figure 1 we illustrate a block diagram for our ICL approach to claim normalization. As a preparatory step, we use a vector database ChromaDB to store training examples and retrieve relevant samples for query input. ChromaDB utilizes the *all-MiniLM-L6-v2* [13] model from Sentence Transformers as its default embedding function, which generates contextualized and semantically meaningful sentence embeddings. These embeddings capture sentence-level semantics and are well suited for similarity search tasks. We used cosine distance as the dissimilarity measure to retrieve top-ranked examples from the vector store for any given input and applied the FICL and AICL approach accordingly.

4.3.1. Fixed In-Context Learning (FICL)

FICL approach inserts a fixed number (K) of examples selected from the training set based on their similarity to the input. The most similar K examples are injected into the prompt (Figure 7) to generate a normalized claim. Formally, the posterior probability of generating the true claim can be expressed as

$$P(y \mid x, k) = f(x, E_k(x); \phi_{\text{LLM}}) \tag{1}$$

where x is input text, $E_k(x)$ is an example set of K numbers of documents that are most similar to x, and ϕ_{LLM} are the decoder parameters of the pretrained LLM.

We ran the generation of normalized claims using FICL using different values of K in the range of 0 (Zero-shot) to upper bound value M=9 examples and computed the average METEOR score of the generated claims against the test labels. The results illustrated in Figure 2 and Table 3 will be discussed in Section 5.

4.3.2. Adaptive In-Context Learning (AICL)

The main idea behind the AICL approach is to dynamically determine the number of examples based on similarity metrics between the input and candidate examples, in contrast to fixed size of the example set in FICL. Our implementation of AICL adopts a simple approach instead of a prediction model as proposed by Chandra et al. [10]. Instead of using a trained model to predict the optimal number of examples K, we applied a heuristic based on cosine distance thresholds. For each test instance, we

selected all training examples whose cosine distances to the input are below a predefined threshold. This approach eliminates the need for a dedicated prediction model while maintaining the adaptability of AICL. Formally, the posterior probability of generating normalized claim can be expressed as Equation 2

$$P(y \mid x, \epsilon) = f(x, E_\epsilon(x); \phi_{\text{LLM}}) \tag{2}$$

where ϵ is the cosine distance threshold that is used to filter similar documents to generate an example set. The notable difference here is that while $E_k(x)$ in Equation 1 is set of examples of fixed size K, $E_\epsilon(x)$ returns a set of examples of varying size based on ϵ.

We experimented with cosine distance threshold values ranging from 0.00 to 1.50 in increments of 0.05 (similar to hyperparameter grid search) and evaluated the corresponding average METEOR scores to determine optimal performing threshold. The results using labeled *dev* dataset are illustrated in Figure 3 and Table 4 and discussed in Section 5.

5. Results and Discussion

We present results on the accuracy of the approaches in Table 2. Here, we focus our results on the English, German, French, Spanish, and Portuguese languages, where we report an average METEOR score for the normalization of the test data sets with gold labels. Among the four approaches we have tried, the **Google flan-t5-l** model that was fine-tuned on training data showed superior performance across majority of the languages except for Portuguese, while zero-shot approach performed the worst. Between ICL based methods, the FICL method shows better performance consistently over AICL across all languages, even surpassing fine-tuned method in Portuguese. The performance gap between FICL and AICL is marginally small in Portuguese. Talking about language-specific trends, English and Spanish have higher scores across all approaches, suggesting easier generalization or better training data in these languages. German shows relatively lower overall performance, possibly due to inadequate training dataset. We further analyze the results of the ICL methods using *dev* dataset in the following section.

Table 2
Average METEOR score for claim normalization using different approaches. Fine-tuned method outperforms ICL based methods in majority of languages except Portuguese, in which both FICL and AICL method yield better claim normalization.

Approaches	Average METEOR score by languages				
	English	German	French	Spanish	Portuguese
Zero-shot	0.21	0.15	0.14	0.21	0.20
Fine-tuned model	**0.40**	**0.26**	**0.37**	**0.38**	0.33
FICL (Mistral-7B)	0.39	0.22	0.32	0.36	**0.36**
AICL (Mistral-7B)	0.37	0.2	0.3	0.33	0.35

5.1. Fixed In-Context Learning (FICL)

The more detailed result of the FICL method in the *dev* dataset is presented in Figure 2 and Table 3. The zero-shot baseline achieved an average METEOR score of 0.24. In contrast, FICL attained a peak score of 0.43 when the number of in-context examples K was set to 4 (Table 3), showing an approximate 80% improvement over the baseline. As illustrated in Figure 2, incorporating a small number of relevant examples leads to substantial gains in generation quality. However, further increasing the number of examples beyond this optimal point results in diminishing performance, likely due to the introduction of noise and less relevant contextual information.

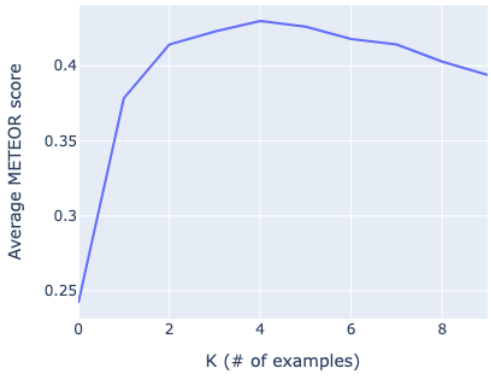

Figure 2: Average METEOR score over K (# of examples used) obtained for English (dev) dataset.

5.2. Adaptive In-Context Learning (AICL)

The AICL configuration achieved its highest performance with an average METEOR score of 0.40 at a cosine distance threshold of 0.75 (Figure 3). This result is comparable to the optimal score observed in the FICL setting, indicating that AICL can approach peak performance without requiring a fixed number of in-context examples.

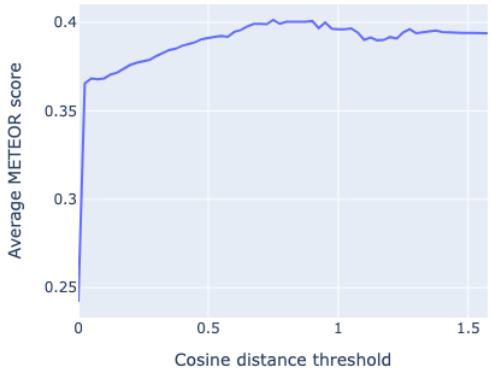

Figure 3: Average METEOR score over varying Cosine distance threshold obtained for English (dev) dataset.

6. Future Work

Among the approaches we have evaluated (as in Table 2), we assume that the margin of improvement for the zero-shot method and the FICL method is narrow due to their reduced model complexity, and therefore we want to prioritize further investigation of the fine-tuned models and the AICL method.

6.1. Fine-tuned model

To further enhance the performance of the fine-tuning approach, several avenues can be explored in future work. One promising direction is to conduct systematic hyperparameter tuning of key training parameters which were not exhaustively optimized in our current setup. Fine-tuning these parameters

could lead to more stable convergence and improved generalization across languages. Additionally, while our experiments were limited to the flan-t5-large model due to resource constraints, larger variants such as flan-t5-xl and flan-t5-xxl offer greater representational capacity and could potentially yield significant performance gains. With access to enhanced computational resources, fine-tuning these larger models on multilingual training data may further improve the quality of claim normalization, especially in high-resource language settings.

6.2. Adaptive In-Context Learning (AICL)

To perform some further analysis on the effectiveness of AICL approach, we compared example set size (K value for each test data point) generated by AICL method to ideal example set size that would have resulted in best claim normalization performance using identical ICL prompt and underlying model, henceforth referred to as oracle method. We compared statistics of example set size of the best performing AICL method and oracle method visually using a heatmap as in Figure 4. We outline following notes for future work in this direction.

- The expectation was to see high datapoint counts along the diagonal of the heatmap Figure 4 if AICL was able to predict the best K values (example set size). We realize that this is not evident from the heatmap. We also visually compared the distribution of predicted K values with that of the best scoring K values (oracle method) using Figure 5 and observed a significant divergence between these two distributions. Hence, this analysis does not support AICL's effectiveness. However, since the average METEOR score improved significantly compared to the zero-shot baseline as in Figure 3, we maintain our confidence in the approach and believe it deserves further investigation.
- We also notice a region in the heatmap Figure 4 (bright yellow region top-left) that represents significant number of instances when including semantically dissimilar examples seem to also have positive effect in claim normalization which is counter intuitive. It may hint that there may exist a mechanism other than semantic similarity that could be used for example selection and improved performance, hence the cases worth investigating.
- Due to constraints in time and computational resources, we were unable to comprehensively evaluate the effectiveness of larger or more advanced language models for claim normalization. However, in preliminary testing using a limited set of samples, we observed that the use of Qwen-8B, a model known for its superior reasoning capabilities and instruction-following performance compared to Mistral-7B, led to a notable improvement in the METEOR scores of generated claims. Based on these encouraging results, we plan to further evaluate the AICL method using more advanced models such as DeepSeek-R1, Qwen3-32B and Llama 4 models in future work.

7. Conclusion

In this study, we evaluated four approaches to claim normalization in a multilingual setting: zero-shot prompting, fine-tuning, FICL, and AICL. Among these, the fine-tuning-based method yielded the highest performance across all evaluated languages except for Portuguese, followed closely by FICL and AICL methods. The zero-shot approach consistently underperformed relative to the others.

Our experiments demonstrate the effectiveness of fine-tuning LLM such as Flan-T5, particularly when adequate training data is available. However, we recognize that the fine-tuning approach can be further enhanced through systematic hyperparameter tuning and the use of more capable pre-trained models.

Although AICL did not outperform FICL in our current setup, we believe it remains a promising approach. Its dynamic example selection mechanism enables context-aware prompting without additional model training. Moreover, AICL's performance, while slightly behind FICL, surpassed the zero-shot baseline by a significant margin. Our analysis reveals opportunities for improving example selection

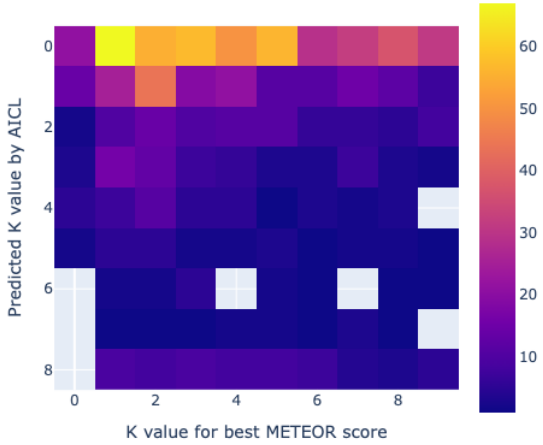

Figure 4: Heatmap showing distribution of test instance count over predicted K (# of examples) by AICL method and K by oracle model (best outcome).

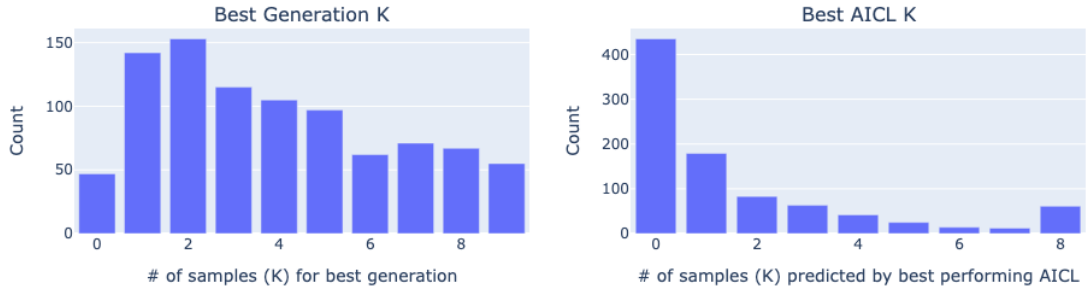

Figure 5: Histirograms showing distribution of K (# of examples) by oracle method (right) and distribution of K by AICL method (left).

strategies—potentially beyond semantic similarity—to boost AICL's effectiveness. These insights, along with encouraging early results from more advanced models, motivate further exploration of AICL using larger architectures such as DeepSeek-R1, Qwen3-32B, and LLaMA 4 in future work.

Declaration on Generative AI

During the preparation of this work, the author(s) used ChatGPT and Overleaf in order to perform grammatical refinements. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] M. Sundriyal, T. Chakraborty, P. Nakov, From chaos to clarity: Claim normalization to empower fact-checking, in: Findings of the Association for Computational Linguistics: EMNLP 2023, 2023.

[2] M. Sundriyal, T. Chakraborty, P. Nakov, Overview of the CLEF-2025 CheckThat! lab task 2 on claim normalization, ????

[3] A. Vlachos, S. Riedel, Fact checking: Task definition and dataset construction, in: Proceedings of the ACL 2014 workshop on language technologies and computational social science, 2014, pp. 18–22.

[4] J. Thorne, A. Vlachos, Automated fact checking: Task formulations, methods and future directions, in: Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018), Association for Computational Linguistics, 2018, pp. 3346–3359. doi:10.18653/v1/C18-1283.

[5] N. Hassan, G. Zhang, F. Arslan, J. Caraballo, D. Jimenez, S. Gawsane, S. Hasan, M. Joseph, A. Kulkarni, A. K. Nayak, V. Sable, C. Li, M. Tremayne, Claimbuster: The first-ever end-to-end fact-checking system, Proceedings of the VLDB Endowment 10 (2017) 1945–1948. doi:10.14778/3137765.3137815.

[6] A. Barrón-Cedeño, F. Alam, J. M. Struß, P. Nakov, T. Chakraborty, T. Elsayed, P. Przybyła, T. Caselli, G. Da San Martino, F. Haouari, et al., Overview of the clef-2024 checkthat! lab: check-worthiness, subjectivity, persuasion, roles, authorities, and adversarial robustness, in: International Conference of the Cross-Language Evaluation Forum for European Languages, Springer, 2024, pp. 28–52.

[7] P. R. Aarnes, V. Setty, P. Galuščáková, Iai group at checkthat! 2024: Transformer models and data augmentation for checkworthy claim detection, in: Notebook for the CheckThat! Lab Task 1 at CLEF 2024, CLEF CheckThat'24, CEUR Workshop Proceedings, Grenoble, France, 2024. CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024.

[8] V. Setty, Surprising efficacy of fine-tuned transformers for fact-checking over larger language models, in: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '24, 2024, p. 2842–2846.

[9] V. Setty, Factcheck editor: Multilingual text editor with end-to-end fact-checking, in: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '24, 2024, p. 2744–2748.

[10] M. Chandra, D. Ganguly, I. Ounis, One size doesn't fit all: Predicting the number of examples for in-context learning, in: Advances in Information Retrieval, 2025.

[11] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tai, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, A. Castro-Ros, M. Pellat, K. Robinson, D. Valter, S. Narang, G. Mishra, A. Yu, V. Zhao, Y. Huang, A. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, J. Wei, Scaling instruction-finetuned language models, J. Mach. Learn. Res. (2024).

[12] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, J. Mach. Learn. Res. (2020).

[13] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, M. Zhou, Minilm: deep self-attention distillation for task-agnostic compression of pre-trained transformers, in: Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20, 2020.

A. Prompts used for Zero-shot and ICL experiments

```
You are a helpful AI assistant.  Given a noisy and unstructured social media
post, rewrite it as a simple and concise statement.

Produce a concise statement for the following post (delimited by ###).
The original language of the post is {language}.

###
{post}
###

Always produce a valid JSON string as a final output using the format below.

{{
  "normalized_claim":   <generated  normalized  claim  translated  in  {language}
language>
}}
```

Figure 6: Prompt used for zero-shot model. The prompt is curated to be clear, concise with delimited input section and expectations on output format clearly defined.

```
You are a helpful AI assistant.  Given a noisy and unstructured social media
post, rewrite it as a simple and concise statement.

Below are some examples of the task intended with input post and expected outcome.

————- Examples ————-

{examples}

—— End of Examples ——-

Produce a concise statement for the following post (delimited by ###).
The original language of the post is {language}.

###
{post}
###

Always produce a valid JSON string as a final output using the format below.

{{
  "normalized_claim":   <generated  normalized  claim  translated  in  {language}
language>
}}
```

Figure 7: Prompt used for ICL model. The in-context examples are injected into the prompt designed in 6.

B. Average METEOR scores obtained for Zero-shot and ICL experiments

Table 3
Average METEOR score of FICL model for different values of K (# of examples). Best achieved score is 0.429 for K=4.

K (# of examples)	Average METEOR score
0	0.241939
1	0.378356
2	0.414237
3	0.422950
4	**0.429926**
5	0.426142
6	0.417937
7	0.414304
8	0.402832
9	0.393938

Table 4
Average METEOR score of AICL model for different values of Cosine Distance threshold. Best achieved score is 0.400828 for threshold of 0.9.

Cosine Distance Threshold	Average METEOR Score
0.00	0.241939
0.05	0.368395
0.10	0.368404
0.15	0.371689
0.20	0.376016
0.25	0.377882
0.30	0.381005
0.35	0.384421
0.40	0.386896
0.45	0.388922
0.50	0.391500
0.55	0.392321
0.60	0.394702
0.65	0.397776
0.70	0.399228
0.75	**0.401437**
0.80	0.400422
0.85	0.400107
0.90	0.400828
0.95	0.399999
1.00	0.396168
1.05	0.396700
1.10	0.390203
1.15	0.389940
1.20	0.391778
1.25	0.394388
1.30	0.393908
1.35	0.394985
1.40	0.394675
1.45	0.394162
1.50	0.394125
1.55	0.393938