The limits of Italian in Reasoning Tasks

Leonardo Ranaldi[1,2], Federico Ranaldi[2], Giulia Pucci[3], Elena Sofia Ruzzetti[2] and Fabio Massimo Zanzotto[2]

[1]*School of Informatics, University of Edinburgh, UK.*
[1]*Università degli Studi Roma "Tor Vergata", Roma, Italy.*
[3]*Department of Computing Science, University of Aberdeen, UK.*

Abstract
Earlier works have been showing the efficacy of *reasoning methods* in eliciting step-wise reasoning of large language models (LLMs) by operating via in-context demonstrations. These strategies, exemplified by Chain-of-Thought (CoT) and Program-Aided Language Models (PAL), have been shown to reason well in monolingual contexts, primarily in English. However, there has been limited investigation into their capabilities in other languages, especially Italian.

To gain a deeper understanding of the role of *reasoning methods*, we propose a multidimensional analysis tailored to Italian, focusing on arithmetic and symbolic reasoning tasks. Our findings indicate that the effectiveness of *reasoning methods* varies significantly beyond English. Expressly, CoT, which relies on natural language demonstrations, is limited to English. Conversely, the structured nature of PAL in-context demonstrations facilitates multilingual comprehension, enabling LLMs to generate programmatic answers in Italian as well. Finally, for a more complete overview, we observe that additional alignment methods do not improve downstream performances; in contrast, in some cases, they restrict the abilities of the original models.

Keywords
Large Language Models, Reasoning Methods, Multilingual Reasoning,

1. Introduction

Large language models (LLMs) are able to tackle tasks using prompts formed by structured patterns, a process known as in-context learning [1]. This method allows the models to solve tasks without modifying their underlying parameters, relying solely on the provided inputs. The success of in-context learning has consequently heightened interest in analysing the factors that influence its effectiveness [2, 3, 4].

Regarding *reasoning methods*, two effective strategies have emerged: Chain-of-Thought (CoT) [5, 6] and Program-Aided Language Models (PAL) [7, 8]. CoT decomposes a reasoning task into a series of intermediate steps using natural language, making it more general and human-understandable. In contrast, PAL employs Python functions to provide reasoning solutions, with its step-by-step programming approach leading to more systematic and structured reasoning.

Although earlier research primarily showcased the functioning of reasoning methods in English, recent studies have expanded to explore multilingual approaches. Shi et al. [9] shown that the effectiveness of CoT rationales is limited to the languages most represented in LLMs pre-training data. Huang et al. [10] addressed the problem by proposing prompting mechanisms that translate the problem into English, while Ranaldi et al. [11] elicit multi- and cross-lingual alignments for enabling reasoning, or Ranaldi et al. [12] self-correction mechanisms. The focus is limited to proposing performance solutions for a few languages, leaving behind the study of the role and the impacts of *languages* such as Italian.

In this paper, we conduct an in-depth study to evaluate the role of reasoning methods in **Italian**. Taking previous work a step further, we study the operation of reasoning methods by analysing the effects of different types of *reasoning methods* on LLMs' Italian reasoning capabilities. This leads to the main research questions of this paper: *(i)* What role do natural language and structured in-context demonstrations play in reasoning planning in *Italian*? *(ii)* What are the impacts and limits of natural language demonstrations? *(iii)* Do *Italian-aligned* and *Italian-centred* models respond differently to reasoning methods?

To answer these questions, we operate via `CoT` and `PAL` (shown in Table 1 and Table 2). For multilingual CoT, we use natural language demonstrations both in English and in Italian following Shi et al. [9]. Instead, for PAL, we propose a novel method by extending the original in English [7]. We use reasoning tasks covering mathematical, commonsense reasoning, and natural language inference tasks in original versions (English) and adapted to Italian (resources available). These tasks are MGSM [9] and MSVAMP [13], which consist of mathematical reasoning problems, and XCOPA [14], PAWS-X [15] and XLNI [16] which consist of commonsense reasoning and

CLiC-it 2024: Tenth Italian Conference on Computational Linguistics, Dec 04 — 06, 2024, Pisa, Italy
[name].[surname]@ed.ac.uk (L. Ranaldi);
[name].[surname]@uniroma2.it (F. Ranaldi);
[name].[surname]@abdn.ac.uk (G. Pucci)

natural language inference.

Finally, we select a range of different LLMs, we employ GPTs [17] models for the results obtained in multilingual tasks, Phi-3 [18], and Mixtral [19] for the results obtained in Italian benchmarks, different versions of Llama-2 and Llama-3 [20] (adapted version for Italian, i.e., Llamantino-2 and -3 [21, 22]), EuroLLM [23] and finally two Italian-centered LLMs for the improvements achieved by smaller-scale versions. We operate using the original models, and we propose aligned versions using state-of-the-art instruction-tuning methods based on synthetic data [24] transferred for multilingual cases [25, 26].

The main contribution and findings of our paper are:

- *Reasoning methods* improve performance in Italian reasoning tasks as well as in English. However, although both methods bring tangible benefits, several limitations emerge in the natural language demonstrations employed in CoT. On the other side of the coin, we observe that the structured reasoning demonstrations (i.e., PAL) elicit the models to plan the solution in a more modularised way. Consequently, this benefits the final performance in both English and non-English tasks.

- We display the positive impact of structured in-context demonstrations on solution planning in Italian. We then demonstrate that since structured reasoning demonstrations are less ambiguous than natural language, they are more adaptable for math reasoning tasks and have a more noticeable impact in more articulate languages such as Italian.

- Finally, we show that the different LLMs analyzed in our contribution are able to understand problems in both English and Italian. However, performance in English is higher despite different approaches used to equate Italian and English proficiency. This reveals that the limitation is not derived from proficiency in a specific language but rather from the language's intrinsic difficulty

To the best of our knowledge, this is the first work that investigates the impact of reasoning methods for the Italian and demonstrates how these strategies can consistently boost LLMs' performance, equipping them with the ability to generate step-wise explanatory reasoning for their predictions. We share the data used at the following link.

2. Reasoning Methods

In-context reasoning methods elicit large language models (LLMs) in delivering step-wise reasoned answers, as presented in §2.1. These methods demonstrate their functionality in several tasks, but evaluations and further studies are primarily conducted in English, leaving other languages unexplored (§2.2). To this end, we propose a methodical study of the effect of reasoning methods beyond English, mainly focusing on Italian (§2.3).

2.1. In-context Learning

Techniques like Chain-of-Thought (CoT) prompting [6] and Program-Aided Language Models (PAL) [7] have improved LLMs' performances by encouraging the generation of intermediate reasoning steps. However, while CoT explanations are not always faithful to the actual reasoning process of the model, with final answers that may not logically follow from the reasoning chain, the structured nature of PAL limits ambiguities and leads the LLMs to deliver structured generations.

2.2. Multilingual Reasoning

Earlier research studied the performances of CoT prompting in different languages. Shi et al. [9] tested the effectiveness of native in-context CoT that are rationales in a specific language (`Native-CoT` in Table 1). Qin et al. [27], inspired by [10] and [28], proposed two-step CoT prompting. Finally, Ranaldi et al. [12] proposed a prompt-based self-correction strategy. However, these studies have focused on demonstrating the performance of CoT and derived methods on large English-focused LLMs. Thus, previous works left a gap in the study of the type of multilingual demonstrations and their impacts and effects on reasoning on different scales of LLMs.

```
Q: Roger ha 5 palline da tennis. Ha comprato altre 2 lattine di palline da tennis. Ogni barattolo contiene 3 palline da tennis. Quante palline da tennis ha ora?
A: Roger inizia con 5 palline. 2 barattoli da 3 palline da tennis ciascuno fanno 6 palline da tennis. 5 + 6 = 11. La risposta è 11.
- - - - - - - - - - - - - - - - - - - -
Q: Leah ha 32 pezzi di cioccolato e sua sorella 42 pezzi. Se hanno mangiato 35 pezzi, quanti pezzi sono rimasti?
A:
```

Table 1
Native Chain-of-Thought (**Native-PAL**) adapted to Italian case (for simplicity, we have reduced the shot, but the original is 6-shot). The in-context question and the rationales are in the specific language (Italian in our case).

2.3. Reasoning in Italian

We take the next step by proposing an in-depth evaluation that studies the effect of in-context demonstrations used in the *reasoning methods*. Hence, we conduct our analysis on different LLMs chosen by family, capabilities, and scope of construction (§3.2) with reasoning tasks (§3.1). The goal is to examine the impact of various types of demonstrations in Italian, addressing the limitations and enhanced functionality these methods can offer.

Our experiments explore the following key points: *a)* constructing robust evaluation by extending PAL (see Table 2) and applying Italian CoT methods on different models using carefully designed benchmarking tasks; *b)* investigating the effects of in-context demonstrations; *c)* analysing the varying effects of in-context reasoning methods across different models (e.g., models without any further adaptation, and models adapted for the Italian language).

PAL beyond English To extend multilingual evaluation to the PAL reasoning method, we propose a specially constructed language-specific version (showed in the following table) by transferring the prompts proposed in [9] into programs-like demonstrations as done in [7].

```
Q: Roger ha 5 palline da tennis. Ha
comprato altre 2 lattine di palline da
tennis. Ogni barattolo contiene 3 palline
da tennis. Quante palline da tennis ha ora?
A: # Roger ha 5 palline da tennis.
    tennis_balls = 5
# compra 2 lattine, ciascuna ha 3 palline
da tennis
    bought_balls = 2 * 3
# Le palline totali sono
    answer = tennis_balls + bought_balls
# La risposta è 11
- - - - - - - - - - - - - - - - - - - - - -
Q: Leah ha 32 pezzi di cioccolato e sua
sorella 42 pezzi. Se hanno mangiato 35
pezzi, quanti pezzi sono rimasti?
A:
```

Table 2
Native Program-Aided Language Models (**Native-PAL**) (we reported one-shot as in Table 1). The in-context questions and the demonstrations are in the native language.

3. Experimental setup

3.1. Data

We introduce five different reasoning tasks: MGSM [9], MSVAMP [13], XNLI [16], and PAWS-X [15], XCOPA [14]; they have been constructed for multilingual evaluations and are described in detail in Appendix 7.

3.2. Models

We select LLMs based on performance and the purpose of the construction. These models are best exemplified by the GPT [17] and Llama-2 and -3 [20] families for the performances shown in multilingual reasoning tasks [9], two models from the Mistral family [19], EuroLLM[1] [23] and Phi-3 [18] for the proficiency shown in the Italian leaderboard. Finally, discerning between the training types, we select Italian-aligned models (Llamantino-2 [21] and Llamantino-3 [22]) and Italian-centred models (modello-Italia, Minerva-3b, -1b). GPT-3.5 is used via API, while the other models are available in open-source format. Appendix 12 describes the parameters and versions used in detail. (We released data & code at the following link).

3.3. Prompting & Evaluation

We operate in two ways concerning mathematical and understanding & commonsense tasks. For mathematical tasks, we align the original CoT and PAL to Italian. We use `Native-CoT` [9] (Table 1) and adapted method proposed in [27] (Appendix 10). Concerning PAL, we introduce Italian demonstrations as in Table 2. For understanding and commonsense tasks, we define input templates that lead LLMs to follow the instructions and aid generation. We construct prompts following [29], using the CoT prompting method to elicit multi-step generations. Finally, we evaluate performance using the accuracy score. Hence, we measure the exact match between generated outputs and labels[2]. We maintain the generation temperatures as recommended in the official papers. For the GPT-3.5, we use the API, while for the others, we used versions available on huggingface (in Appendix 12).

[1] **NB** we identify this model as Italian-centred even though it has been pre-trained on different European languages in the same way [23].

[2] We extract target labels from the generated answers using regular expressions before calculating the exact match. For each task, we use *Instruction Templates* to guide the model to stable generations and facilitate evaluation.

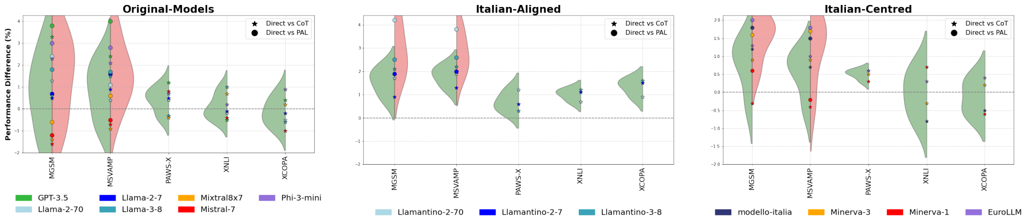

Figure 1: Performance difference between accuracies obtained by using `Direct` prompting and `Native-CoT` (marker) and `Native-PAL` (marker). Each point represents the performance across models obtained adapting *reasoning method* to a specific language (i.e., `Native` prompting). In Appendices 14, 15, 16, and 17 are reported detailed results.

4. Results & Discussions

Large language models (LLMs) benefit from *reasoning methods* in English and in Italian as well. As discussed in §4.1, the in-context demonstrations beyond English elicit the LLMs to deliver multilingual reasoned answers; however, the operation differs depending on the type of method.

Although demonstrations lead the models to generate more robust answers, improving Italian as well, the operation of these techniques appears to be effective only in some models. As analysed in §4.2, in-context rationales in natural language have a different effect. On the other side of the coin, structured program-of-thoughts demonstrations lead the models to more stable generations. Hence, the impact of in-context demonstrations varies according to the quality and quantity of rationales and the scale of model parameters (§4.3).

Finally, in §4.4, we examine the effects of alignment approaches by discerning the factors that influence the generation of the final response and highlighting the matter of native language demonstrations.

4.1. Reasoning in Italian

In-context reasoning methods empower the LLMs' multilingual performances in arithmetic and symbolic reasoning tasks. Figure 1 shows the differences between `Native-CoT` and `Native-PAL`, and the baselines (`Direct`). The use of in-context Italian demonstrations brings clear benefits. GPT-3.5 and Llama-based models (Llama2-70 and Llamantino3) obtain noticeable benefits from `Native-based` prompting approaches (complete results in Appendix 14). Although these LLMs benefit the most from introducing reasoning methods in the prompting stage, further improvements are observable even in LLMs with fewer parameters (i.e., EuroLLM, Phi-3, Llama-2-7, and Llama3-8 as well adapted versions Llamantino-2 and -3, complete results in Appendices 15, 16). These results demonstrate the sensitivity of Italian in-context prompting in understanding and commonsense reasoning (Appendix 17). However, although the averages are mainly positive, some phenomena emerge, such as differences (the baseline `Direct` outperforms the reasoning method) and a disparity between `CoT` and `PAL` between *Original-* and *Italian-Aligned* models. Specifically, *(i)* `PAL` (⋆) outperforms `CoT` (●) in Figure 1 and *(ii)* the *Italian-Aligned* models outperform the *Original-Model* in Italian task but not in English. To understand these dynamics in depth in §4.2, we explore how the demonstration structure impacts the models' generations.

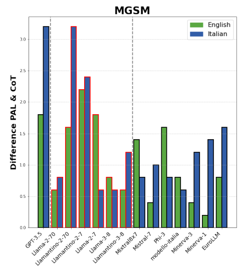

Figure 2: Difference between `PAL` and `CoT` (highlighted the original and adapted models)

4.2. Natural Language Effects

The effect of the reasoning method relies on the solution strategy. Structured in-context demonstrations in a program-like manner are more effective than natural language rationales. Figure 2 displays that the differences between `Native-PAL` and `Native-CoT` are consistently positive. Moreover, the Italian-Aligned models (i.e., Llamantino-based) obtain better results of original models in Italian tasks when `Native-PAL` is used. Since the natural language of in-context rationales does not provide the same benefits as PAL, we examined the generations delivered to investigate the origin of the differences.

The results indicate that even though the CoT in-context demonstrations in the Italian natural language are the same as those in English, the generations have different structures (Appendix 9, Table 7). In-depth, a relationship emerges between performance and the average number of steps required to get correct answers. The number of $Hops$, i.e., the steps to reach the final solution, represented by natural language sentences, are on average between 2 and 5 for the Italian answers and around 3 and 5 for English; in PAL, they are concentrated around 3 and 4. This shows that natural language, especially Italian, rich in intricate linguistic structures, is not the best for solving mathematical, symbolic tasks. In contrast, PAL seems more appropriate due to its rigid structure and better support for generative reasoning passages.

4.3. Demonstrations Impacts

In-context demonstrations play a key role in complex tasks because they promote reasoning, as discussed in §4.1. We investigated the performance trend as in-context demonstrations increased, repeating the previous experiments focusing on MGSM using zero- from 6-shots. The results show that the impact of in-context demonstrations across the languages is related to the quality and quantity of demonstrations. A distinction emerges between models and the number of de facto useful demonstrations. GPT-3.5 with 4-shots achieves results comparable to 6-shots (average accuracies in Figure 6). This balance does not occur in Llama-based and Mixtral, which underperforms as in-context demonstrations increase. Finally, the smaller models have conspicuous improvements as the number of demonstrations increases.

4.4. Language of Reasoning makes the difference

Multilingual in-context demonstrations aid LLMs in applying solution strategies; however, the language used to reason matters. By eliciting LLMs to deliver multi-step English answers, we observed significant improvements in accuracy. Complementing previous work, we used two strategies: *(i)* in-context demonstrations of reasoning answers in a specific language (`Native-method`). *(ii)* the same in-context setting and then elicit the model to provide the solution in English (`Cross-method`). As in Table 3, the `Cross-methods` provide tangible benefits both in PAL and CoT. These latter results emphasized the LLMs' understanding and production abilities.

Model		Δ**MGSM**	Δ**MSVAMP**	Δ**XCOPA**	Δ**XNLI**	Δ**PAWS-X**
`GPT-3.5`	**CoT**	+4.8	+5.2	+0.6	+4.2	+3.6
	PAL	+3.8	+2.7	-	-	-
`Llama-70`	**CoT**	+3.4	+2.0	+4.6	+5.4	+1.9
	PAL	+2.8	+2.7	-	-	-
`Llama`$_{IT}$`-70`	**CoT**	+3.6	+0.8	+0.3	+3.1	-0.4
	PAL	+2.6	+4.0	-	-	-
`Llama-7`	**CoT**	+4.2	+2.3	+1.1	+3.6	+0.8
	PAL	+3.4	+2.7	-	-	-
`Llama`$_{IT}$`-7`	**CoT**	+2.0	+0.5	+1.8	+1.3	-0.6
	PAL	+2.4	+1.4	-	-	-
`Llama-8`	**CoT**	+3.2	-0.1	+2.3	+3.2	+0.8
	PAL	+4.8	+1.9	-	-	-
`Llama`$_{IT}$`-8`	**CoT**	+1.0	+1.9	+0.4	+2.3	+1.2
	PAL	+1.2	+2.3	-	-	-
`mod-italia`	**CoT**	+2.2	+2.5	+0.0	+3.1	+1.7
	PAL	+3.2	+1.7	-	-	-
`Minerva-3`	**CoT**	+2.2	+1.3	-0.2	-0.9	+0.6
	PAL	+3.1	+2.1	-	-	-
`EuroLLM`	**CoT**	+0.2	+1.6	+0.8	-0.2	+0.2
	PAL	+1.2	+0.3	-	-	-

Table 3
Differences between `Cross-` and `Native-based`. *(`Llama`$_{IT}$ are `Llamantino` models)

5. Findings & Future Works

We investigate the impact that *reasoning methods* cause on final performance by expanding the study about the role and the limits of them in *Italian*. The main findings and tangible recommendations can be outlined as follows: ***a)*** Reasoning methods work in Italian as well; however, there emerges a difference between rationales-based methods (CoT) and program-like approaches (PAL). ***b)*** The nature of natural language demonstrations used in CoT does not fit best with rich languages such as Italian. Instead, PALs' programme structure limits ambiguity by improving the ability to deliver reasoning in English and Italian. ***(c)*** Consequently, this analysis recommends operating through structured in-context rationale instead of using natural language when interacting with LLMs, especially when dealing with complex contexts such as reasoning. In the future, we would like to investigate the internal dynamics that support the causal generations of LLMs to identify gaps and improve multilingual generative capabilities [30] by exploiting alignment [24] or self-refining approaches [31]. However, at the same time, contamination data issues [32, 33, 34]

6. Conclusion

The advances of *reasoning methods* emerge beyond the English. Our analysis shows that properly elicited LLMs can deliver reasoned answers in Italian as well. By operating via CoT and PAL, we revealed that in-context demonstrations play a strategic role in improving performance in direct proportion to their quality and quantity. Our research highlights the need for a customised strategy for employing reasoning methods for LLMs. It supports the demand for a reasonable combination of model scale, reasoning technique, and strategic use of in-context learning to elicit the prospect of multilingual LLMs.

Acknowledgements

This work was funded by UK Research and Innovation (UKRI) under the UK government's Horizon Europe funding guarantee grant number 10039436 and PRIN 2022 Project - Class-tAIs CUP: E53D230081000.

References

[1] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are few-shot learners, 2020. arXiv:2005.14165.

[2] O. Rubin, J. Herzig, J. Berant, Learning to retrieve prompts for in-context learning, in: M. Carpuat, M.-C. de Marneffe, I. V. Meza Ruiz (Eds.), Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Seattle, United States, 2022, pp. 2655–2671. URL: https://aclanthology.org/2022.naacl-main.191. doi:10.18653/v1/2022.naacl-main.191.

[3] J. Zhao, Y. Xie, K. Kawaguchi, J. He, M. Xie, Automatic model selection with large language models for reasoning, in: H. Bouamor, J. Pino, K. Bali (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2023, Association for Computational Linguistics, Singapore, 2023, pp. 758–783. URL: https://aclanthology.org/2023.findings-emnlp.55. doi:10.18653/v1/2023.findings-emnlp.55.

[4] Y. Zhang, S. Feng, C. Tan, Active example selection for in-context learning, 2022. arXiv:2211.04486.

[5] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, Y. Iwasawa, Large language models are zero-shot reasoners, 2023. arXiv:2205.11916.

[6] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, D. Zhou, Chain-of-thought prompting elicits reasoning in large language models, 2023. arXiv:2201.11903.

[7] L. Gao, A. Madaan, S. Zhou, U. Alon, P. Liu, Y. Yang, J. Callan, G. Neubig, Pal: Program-aided language models, arXiv preprint arXiv:2211.10435 (2022).

[8] W. Chen, X. Ma, X. Wang, W. W. Cohen, Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks, 2023. arXiv:2211.12588.

[9] F. Shi, M. Suzgun, M. Freitag, X. Wang, S. Srivats, S. Vosoughi, H. W. Chung, Y. Tay, S. Ruder, D. Zhou, D. Das, J. Wei, Language models are multilingual chain-of-thought reasoners, 2022. arXiv:2210.03057.

[10] H. Huang, T. Tang, D. Zhang, W. X. Zhao, T. Song, Y. Xia, F. Wei, Not all languages are created equal in llms: Improving multilingual capability by cross-lingual-thought prompting, 2023. arXiv:2305.07004.

[11] L. Ranaldi, G. Pucci, A. Freitas, Empowering cross-lingual abilities of instruction-tuned large language models by translation-following demonstrations, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Findings of the Association for Computational Linguistics ACL 2024, Association for Computational Linguistics, Bangkok, Thailand and virtual meeting, 2024, pp. 7961–7973. URL: https://aclanthology.org/2024.findings-acl.473. doi:10.18653/v1/2024.findings-acl.473.

[12] L. Ranaldi, G. Pucci, F. Ranaldi, E. S. Ruzzetti, F. M. Zanzotto, A tree-of-thoughts to broaden multi-step reasoning across languages, in: K. Duh, H. Gomez, S. Bethard (Eds.), Findings of the Association for Computational Linguistics: NAACL 2024, Association for Computational Linguistics, Mexico City, Mexico, 2024, pp. 1229–1241. URL: https://aclanthology.org/2024.findings-naacl.78. doi:10.18653/v1/2024.findings-naacl.78.

[13] N. Chen, Z. Zheng, N. Wu, M. Gong, Y. Song, D. Zhang, J. Li, Breaking language barriers in multilingual mathematical reasoning: Insights and observations, 2023. arXiv:2310.20246.

[14] E. M. Ponti, G. Glavaš, O. Majewska, Q. Liu, I. Vulić, A. Korhonen, XCOPA: A multilingual dataset for causal commonsense reasoning, in: B. Webber, T. Cohn, Y. He, Y. Liu (Eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 2362–2376. URL: https://aclanthology.org/2020.emnlp-main.185. doi:10.18653/v1/2020.emnlp-main.185.

[15] Y. Yang, Y. Zhang, C. Tar, J. Baldridge, PAWS-X: A cross-lingual adversarial dataset for paraphrase identification, in: K. Inui, J. Jiang, V. Ng, X. Wan (Eds.), Proceedings of the 2019 Conference on Empirical Methods in Natural Language

Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3687–3692. URL: https://aclanthology.org/D19-1382. doi:10.18653/v1/D19-1382.

[16] A. Conneau, R. Rinott, G. Lample, A. Williams, S. Bowman, H. Schwenk, V. Stoyanov, XNLI: Evaluating cross-lingual sentence representations, in: E. Riloff, D. Chiang, J. Hockenmaier, J. Tsujii (Eds.), Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 2475–2485. URL: https://aclanthology.org/D18-1269. doi:10.18653/v1/D18-1269.

[17] OpenAI, Gpt-4 technical report, 2023. arXiv:2303.08774.

[18] M. Abdin, S. A. Jacobs, A. A. Awan, J. Aneja, A. Awadallah, H. Awadalla, N. Bach, A. Bahree, A. Bakhtiari, J. Bao, H. Behl, A. Benhaim, M. Bilenko, J. Bjorck, S. Bubeck, Q. Cai, M. Cai, C. C. T. Mendes, W. Chen, V. Chaudhary, D. Chen, D. Chen, Y.-C. Chen, Y.-L. Chen, P. Chopra, X. Dai, A. D. Giorno, G. de Rosa, M. Dixon, R. Eldan, V. Fragoso, D. Iter, M. Gao, M. Gao, J. Gao, A. Garg, A. Goswami, S. Gunasekar, E. Haider, J. Hao, R. J. Hewett, J. Huynh, M. Javaheripi, X. Jin, P. Kauffmann, N. Karampatziakis, D. Kim, M. Khademi, L. Kurilenko, J. R. Lee, Y. T. Lee, Y. Li, Y. Li, C. Liang, L. Liden, C. Liu, M. Liu, W. Liu, E. Lin, Z. Lin, C. Luo, P. Madan, M. Mazzola, A. Mitra, H. Modi, A. Nguyen, B. Norick, B. Patra, D. Perez-Becker, T. Portet, R. Pryzant, H. Qin, M. Radmilac, C. Rosset, S. Roy, O. Ruwase, O. Saarikivi, A. Saied, A. Salim, M. Santacroce, S. Shah, N. Shang, H. Sharma, S. Shukla, X. Song, M. Tanaka, A. Tupini, X. Wang, L. Wang, C. Wang, Y. Wang, R. Ward, G. Wang, P. Witte, H. Wu, M. Wyatt, B. Xiao, C. Xu, J. Xu, W. Xu, S. Yadav, F. Yang, J. Yang, Z. Yang, Y. Yang, D. Yu, L. Yuan, C. Zhang, C. Zhang, J. Zhang, L. L. Zhang, Y. Zhang, Y. Zhang, Y. Zhang, X. Zhou, Phi-3 technical report: A highly capable language model locally on your phone, 2024. URL: https://arxiv.org/abs/2404.14219. arXiv:2404.14219.

[19] A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, D. de las Casas, E. B. Hanna, F. Bressand, G. Lengyel, G. Bour, G. Lample, L. R. Lavaud, L. Saulnier, M.-A. Lachaux, P. Stock, S. Subramanian, S. Yang, S. Antoniak, T. L. Scao, T. Gervet, T. Lavril, T. Wang, T. Lacroix, W. E. Sayed, Mixtral of experts, 2024. arXiv:2401.04088.

[20] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, T. Scialom, Llama 2: Open foundation and fine-tuned chat models, 2023. arXiv:2307.09288.

[21] P. Basile, E. Musacchio, M. Polignano, L. Siciliani, G. Fiameni, G. Semeraro, Llamantino: Llama 2 models for effective text generation in italian language, 2023. URL: https://arxiv.org/abs/2312.09993. arXiv:2312.09993.

[22] M. Polignano, P. Basile, G. Semeraro, Advanced natural-based interaction for the italian language: Llamantino-3-anita, 2024. URL: https://arxiv.org/abs/2405.07101. arXiv:2405.07101.

[23] P. H. Martins, P. Fernandes, J. Alves, N. M. Guerreiro, R. Rei, D. M. Alves, J. Pombal, A. Farajian, M. Faysse, M. Klimaszewski, P. Colombo, B. Haddow, J. G. C. de Souza, A. Birch, A. F. T. Martins, Eurollm: Multilingual language models for europe, 2024. URL: https://arxiv.org/abs/2409.16235. arXiv:2409.16235.

[24] L. Ranaldi, A. Freitas, Aligning large and small language models via chain-of-thought reasoning, in: Y. Graham, M. Purver (Eds.), Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, St. Julian's, Malta, 2024, pp. 1812–1827. URL: https://aclanthology.org/2024.eacl-long.109.

[25] L. Ranaldi, G. Pucci, F. M. Zanzotto, Modeling easiness for training transformers with curriculum learning, in: R. Mitkov, G. Angelova (Eds.), Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing, INCOMA Ltd., Shoumen, Bulgaria, Varna, Bulgaria, 2023, pp. 937–948. URL: https://aclanthology.org/2023.ranlp-1.101.

[26] L. Ranaldi, G. Pucci, Does the English matter? elicit cross-lingual abilities of large language models, in: D. Ataman (Ed.), Proceedings of the 3rd Workshop on Multi-lingual Representation Learning (MRL), Association for Computational Linguistics, Singapore, 2023, pp. 173–183. URL: https://aclanthology.org/2023.mrl-1.14. doi:10.18653/v1/2023.mrl-1.14.

[27] L. Qin, Q. Chen, F. Wei, S. Huang, W. Che, Cross-lingual prompting: Improving zero-shot chain-of-thought reasoning across languages, 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. 2695–2709. URL: https://aclanthology.org/2023.emnlp-main.163. doi:10.18653/v1/2023.emnlp-main.163.

[28] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, D. Zhou, Self-consistency improves chain of thought reasoning in language models, 2023. arXiv:2203.11171.

[29] K. Ahuja, H. Diddee, R. Hada, M. Ochieng, K. Ramesh, P. Jain, A. Nambi, T. Ganu, S. Segal, M. Ahmed, K. Bali, S. Sitaram, MEGA: Multilingual evaluation of generative AI, 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. 4232–4267. URL: https://aclanthology.org/2023.emnlp-main.258. doi:10.18653/v1/2023.emnlp-main.258.

[30] L. Ranaldi, G. Pucci, B. Haddow, A. Birch, Empowering multi-step reasoning across languages via program-aided language models, in: Y. Al-Onaizan, M. Bansal, Y.-N. Chen (Eds.), Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Miami, Florida, USA, 2024, pp. 12171–12187. URL: https://aclanthology.org/2024.emnlp-main.678.

[31] L. Ranaldi, A. Freitas, Self-refine instruction-tuning for aligning reasoning in language models, 2024. URL: https://arxiv.org/abs/2405.00402. arXiv:2405.00402.

[32] F. Ranaldi, E. S. Ruzzetti, D. Onorati, L. Ranaldi, C. Giannone, A. Favalli, R. Romagnoli, F. M. Zanzotto, Investigating the impact of data contamination of large language models in text-to-SQL translation, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Findings of the Association for Computational Linguistics ACL 2024, Association for Computational Linguistics, Bangkok, Thailand and virtual meeting, 2024, pp. 13909–13920. URL: https://aclanthology.org/2024.findings-acl.827. doi:10.18653/v1/2024.findings-acl.827.

[33] L. Ranaldi, G. Pucci, Knowing knowledge: Epistemological study of knowledge in transformers, Applied Sciences 13 (2023). URL: https://www.mdpi.com/2076-3417/13/2/677. doi:10.3390/app13020677.

[34] L. Ranaldi, M. Gerardi, F. Fallucchi, Cryptonet: Using auto-regressive multi-layer artificial neural networks to predict financial time series, Information 13 (2022). URL: https://www.mdpi.com/2078-2489/13/11/524. doi:10.3390/info13110524.

[35] L. Ranaldi, G. Pucci, F. Ranaldi, E. S. Ruzzetti, F. M. Zanzotto, Empowering multi-step reasoning across languages via tree-of-thoughts, 2024. arXiv:2311.08097.

[36] R. Li, L. B. Allal, Y. Zi, N. Muennighoff, D. Kocetkov, C. Mou, M. Marone, C. Akiki, J. Li, J. Chim, Q. Liu, E. Zheltonozhskii, T. Y. Zhuo, T. Wang, O. Dehaene, M. Davaadorj, J. Lamy-Poirier, J. Monteiro, O. Shliazhko, N. Gontier, N. Meade, A. Zebaze, M.-H. Yee, L. K. Umapathi, J. Zhu, B. Lipkin, M. Oblokulov, Z. Wang, R. Murthy, J. Stillerman, S. S. Patel, D. Abulkhanov, M. Zocca, M. Dey, Z. Zhang, N. Fahmy, U. Bhattacharyya, W. Yu, S. Singh, S. Luccioni, P. Villegas, M. Kunakov, F. Zhdanov, M. Romero, T. Lee, N. Timor, J. Ding, C. Schlesinger, H. Schoelkopf, J. Ebert, T. Dao, M. Mishra, A. Gu, J. Robinson, C. J. Anderson, B. Dolan-Gavitt, D. Contractor, S. Reddy, D. Fried, D. Bahdanau, Y. Jernite, C. M. Ferrandis, S. Hughes, T. Wolf, A. Guha, L. von Werra, H. de Vries, Starcoder: may the source be with you!, 2023. arXiv:2305.06161.

7. Proposed Task

Dataset	Task	Languages	#Languages
MGSM	mathematical reasoning	Bengali (bn), Chinese (zh), French (fr), Thai (th) German (de), Japanese (jp), Russian (ru), Telugu (te) Spanish (es), Swahili (sw), **English (en)**	11
MSVAMP	mathematical reasoning	Bengali (be), Chinese (zh), French (fr), Thai (th) German (de), Japanese (jp), Russian (ru) Spanish (es), Swahili (sw), **English (en)**	10
XNLI	natural language inference	**English (en)**, German (de), Russian (ru), French (fr), Spanish (es), Chinese (zh), Vietnamese (vi), Arabic (ar), Greek (el), Thai (th), Bulgarian (bg), Urdu (ur), Swahili (sw), Hindi (hi), Turkish (tr)	15
XCOPA	commonsense reasoning	Chinese (zh), **Italian (it)**, Vietnamese (vi), Turkish (tr), Thai (th), Estonian (et), Tamil (ta), Swahili (sw), Haitian (ht), Quechua (qu), Indon. (in)	11
PAWS-X	paraphrase identification	**English (en)**, German (de), Japanese (jp), French (fr), Spanish (es), Chinese (zh), Korean (ko), **Italian (it)**	8

Table 4
Languages present in datasets used in this work. We used the versions released in English and Italian where it was present. For the missing translations (MGSM, MSVAMP, XNLI), we performed a translation step phase GPT-3.5. Translated versions released on the GitHub repository.

Benchmark	#Test	Final Prompt
MGSM	250	Q: {problem}
MSVAMP	1000	Q: {problem}
XCOPA	200	Here is a premise: {premise}. What is the {question}? Help me pick the more plausible option: -choice1: {choice1}, -choice2: {choice2}
XCOPA	200	Data la premessa: {premise}. Quele è la {question}? Aiutami a scegliere l'opzione piu plausibile: -scelta1: {choice1}, -scelta2: {choice2}
XNLI	200	{premise}. Based on the previous passage, is it true that {hypothesis}? Yes, No, or Maybe?
XNLI	200	{premise}. Basandoti sui precedenti passaggi, è vero che {hypothesis}? Sì, No, o Forse?
PAWS-X	200	Sentence 1: {sentence1} Sentence 2: {sentence2} Question: Does Sentence 1 paraphrase Sentence 2? Yes or No?
PAWS-X	200	Frase1: {sentence1} Frase2: {sentence2} La Frase1 parafrasa la Frase2? Sì or No?

Table 5
The column **#Test** denotes the number of instances for each language in the test set proposed by the authors. The constructions of these tasks are derived from translations (manual or automatic) of subsets of the original monolingual versions (in English) as explained in Section 3.1.

8. In-context Demonstrations

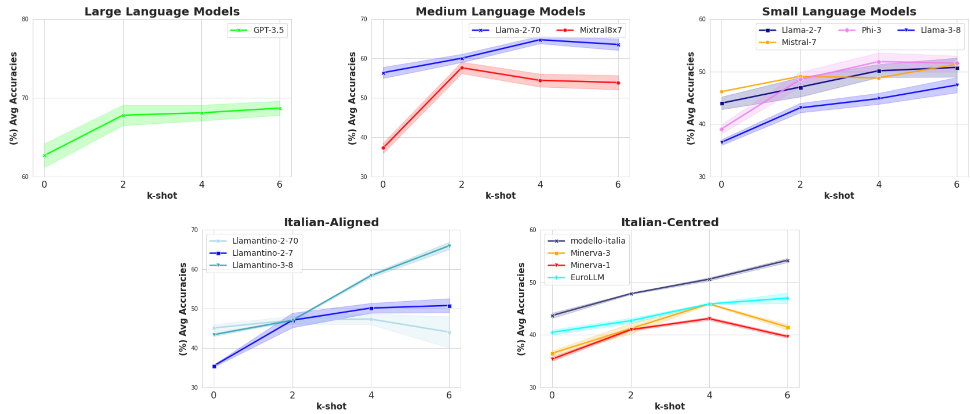

Table 6
Average accuracies on MGSM using methods proposed in (Section 3.3) setting providing in input k-shot demonstrations with k equal to {0,2,4,6}.

9. Natural Language Structure

Analysing the composition of languages in the answers provided by the different models is useful to understand whether a certain model follows the in-context prompts by generating language-specific answers and, if so, what the error rate is. It is important to analyse the composition of the provided answers. To qualitatively estimate the generated responses, we propose the analysis of the phrases present in the responses generated by the models under study. Given an answer A, composed of a set of sentences ($\{s_1, s_2, \ldots, s_n\}$), we define $Hops$ as the number of sentences the models generate to deliver the solution. Since the in-context rationales provided have an average number of 4 $Hops$ (min value 3 and max value 5) [9], they do not include the final keyword *"Answer:"* or *"The answer is:"*, we do not consider the final keyword for a more realistic value as it often repeats the last sentence. Formally, let A be composed of n sentences and represent the final answer. The sum of sentences in A gives the total number of $Hops$. Hence, we compute this value for the generations of models analysed and report results in Table 7.

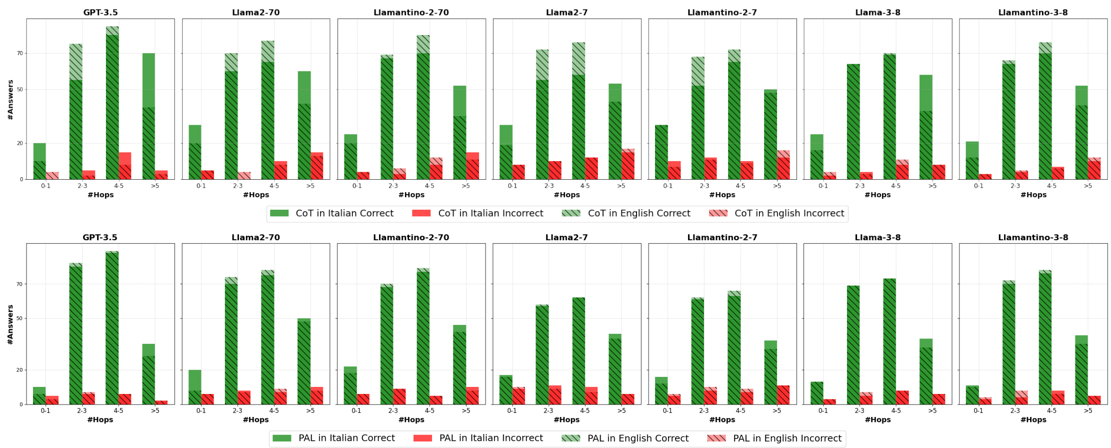

Table 7
Number of $Hops$ generated via CoT and PAL in-context reasoning methods. We describe the concept of $Hops$ in Appendix 9.
*This analysis was performed only on the following models as they consistently provide stable generations.

10. State-of-art Prompting Methods

`Direct` (Question in Chinese without CoT)

Q:： 罗杰有5 个网球。他又买了2 罐网球。每罐有3 个网球。他现在有多少个网球?
A: 11
Q: 利亚有32 块巧克力，她妹妹有42 块。如果她们吃了35 块，她们一共还剩下多少块?
A:

`Native-CoT` (Question and CoT Answer in Chinese)

Q: 罗杰有5 个网球。他又买了2 罐网球。每罐有3 个网球。他现在有多少个网球?
A: 罗杰一开始有5 个球。2 罐各3 个网球就是6 个网球。5 + 6 = 11。答案是11。
Q: 利亚有32 块巧克力，她妹妹有42 块。如果她们吃了35 块，她们一共还剩下多少块?
A:

`En-CoT` (Question in Italian and answer in English)

Q: 罗杰有5 个网球。他又买了2 罐网球。每罐有3 个网球。他现在有多少个网球?
A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. 5 + 6 = 11. The answer is 11.
Q: 利亚有32 块巧克力，她妹妹有42 块。如果她们吃了35 块，她们一共还剩下多少块?
A:

Table 8
Chain-of-Thought as proposed in [9] (for simplicity we have reduced the shot but the original is 6-shot). Given a problem in specific language, the following prompts are `Direct`, `Native-CoT` (without additional languages) and `En-CoT`, the original question in specific language with answers in English.

`Cross-ToT`

```
Simulate the collaboration of {n} mathematicians answering a question in their mother tongue: L1, L2, ... and Ln. They all start Step1 from a separate thought process, step by step, each explaining their thought process. Following Step1, each expert refines and develops their thought process by comparing themselves with others. This process continues until a definitive answer to the question is obtained.
Question: [Question in Language L1]
Answer: [num].
```

Table 9
Cross-ToT prompting [35] that using Tree-of-Thoughts method elicit the model to produce multi-step reasoning processes in different languages.

11. Program-Aided Language Models Prompts

In this paper, as introduced in §3.3, we propose a novel Cross-lingual extension of the Program-Aided Language Models [7] (Cross-PAL) method. The following tables show the prompts used for the final evaluation.

`Program-Aided Language Models (PAL)`

```
Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?
A: Roger started with 5 tennis balls.
   tennis_balls = 5
   2 cans of 3 tennis balls each is
   bought_balls = 2 * 3 tennis balls.
   The answer is
   answer = tennis_balls + bought_balls
   The answer is 11
- - - - - - - - - - - - - - - - - - - - - - - -
Q: Kyle bought last year's best-selling book for $19.50. This is with a 25% discount from the original price. What was the original price?
A:
```

Table 10
This is an example prompt of the PAL method proposed by [7].

$Cross$ **`Program-Aided Language Models`**

```
Q: Michael hat 58 Golfbälle. Am Dienstag hat er 23 Golfbälle verloren. Am Mittwoch hat er 2 weitere verloren. Wie viele Golfbälle hat er Mittwoch am Ende des Tages?
A: Michael hat 58 Golfbälle.
   initial = 58
   Am Dienstag verlor er 23 Golfbälle
   lost_tuesday = 23
   Am Mittwoch verlor er 2 Golfbälle
   lost_wednesday = 2
   Golfbälle abzüglich der verlorenen
    answer = initial - lost_tuesday
     - lost_wednesday
    Die Antwort ist 33
```

Table 11
In Cross-PAL, we use the same setting earlier proposed with PAL demonstrations in the same language of the question.

12. Model and Hyperparameters

In our experimental setting, as introduced in Section 3.2, we propose different LLMs: (i) one model from the GPT family [17]: GPT-3.5 (`gpt-3.5-turbo-0125`); (ii) three models from the Llama-2 family [20]: Llama2-7b, Llama2-70b, Llama-3-8-instruct; (iii) two models of the MistralAI family: Mistral-7b and Mixtral [19]; (iv) finally, Phi-3-mini [36].
In particular, GPTs models are used via API, while for the others, we used versions of the quantized to 4-bit models that use GPTQ (see detailed versions in Table 12)
Furthermore, we have added additional LLMs. These models are three versions of Llama-based models adapted for Italian [21, 22] and three Italian-centered models: modello-Italia, Minerva-3b, and Minerva-1b.
As discussed in the limitations, our choices are related to reproducibility and the cost associated with non-open-source models. We use closed-source API and the 4-bit GPTQ quantized version of the model on 8 48GB NVIDIA RTXA600 GPUs for all experiments performed only in inference.
Finally, the generation temperature varies from $\tau = 0$ of GPT models to $\tau = 0.5$ of Llama2s. We choose these temperatures for (mostly) deterministic outputs, with a maximum token length of 256. The other parameters are left unchanged as recommended by the official resources. We will release the code and the dataset upon acceptance of the paper.

13. Models Vesions

Model	Version
Llama2-7	meta-llama/Llama-2-7b
Llama2-70	meta-llama/Llama-2-70b
Llama3-8	meta-llama/Meta-Llama-3-8B-Instruct
Phi-3-mini	microsoft/Phi-3-mini-128k-instruct
Mistral-7	mistralai/Mistral-7B-Instruct-v0.2
Mixtral8x7	TheBloke/Mixtral-8x7B-Instruct-v0.1-GPTQ
GPT-3.5-turbo	OpenAI API (gpt-3.5-turbo-0125)
Llamantino2-70	swap-uniba/LLaMAntino-2-70b-hf-UltraChat-ITA
Llamantino2-7	swap-uniba/LLaMAntino-2-chat-7b-hf-UltraChat-ITA
Llamantino3-7	swap-uniba/LLaMAntino-3-ANITA-8B-Inst-DPO-ITA
modello-italia	sapienzanlp/modello-italia-9b-bf16
Minerva-3b	sapienzanlp/Minerva-3B-base-v1.0
Minerva-1b	sapienzanlp/Minerva-1B-base-v1.0
EuroLLM	utter-project/EuroLLM-1.7B-Instruct

Table 12
List the versions of the models proposed in this work, which can be found on huggingface.co. We used the configurations described in Appendix 12 in the repositories for each model *(access to the following models was verified on 14 June 2024).

14. Results Arithmetic Reasoning Tasks - English and Italian -

		MGSM			MSVAMP		
Model	**Method**	**en**	**It**	**cross**	**en**	**It**	**cross**
	Direct	80.4	64.0	-	82.7	64.7	-
GPT-3.5	Native-CoT	84.8	66.4	71.2	85.2	69.8	74.0
	Native-PAL	**86.6**	**69.8**	**73.6**	**86.3**	**71.6**	**74.6**
	Direct	70.2	58.4	-	73.7	61.8	-
Llama2-70	Native-CoT	71.8	60.6	64.2	75.3	62.6	64.2
	Native-PAL	**72.4**	**61.2**	**63.0**	**76.9**	**63.0**	**65.7**
	Direct	64.6	53.6	-	68.5	56.9	-
Llama2-7	Native-CoT	67.8	54.2	58.2	69.4	58.1	60.3
	Native-PAL	**69.2**	**55.0**	**58.4**	**70.1**	**58.7**	**61.6**
	Direct	76.4	67.6	-	77.2	68.7	-
Llama3-8	Native-CoT	78.6	69.4	72.6	79.8	69.8	69.7
	Native-PAL	**79.2**	**70.0**	**74.8**	**81.6**	**70.3**	**72.2**
	Direct	76.0	**64.6**	-	**78.0**	66.7	-
Mixtral8x7	Native-CoT	75.4	63.4	62.6	76.3	65.5	66.3
	Native-PAL	**77.2**	64.2	**64.4**	77.8	**67.3**	68.2
	Direct	66.2	**62.8**	-	**67.8**	**62.4**	-
Mistral-7	Native-CoT	66.8	61.0	62.4	66.9	61.5	63.3
	Native-PAL	**67.2**	62.2	**63.0**	67.3	62.1	**64.2**
	Direct	76.8	62.6	-	77.5	63.7	-
Phi-3	Native-CoT	80.4	66.2	72.2	80.3	67.5	74.6
	Native-PAL	**82.0**	**67.4**	**73.0**	**81.0**	**69.4**	**75.5**

Table 13
Accuracies (%) on English and Italian versions of MGSM and MSVAMP using the reasoning methods described in §3.3 (for each model, we reported best performances in **bold**).

15. Results Arithmetic Reasoning Tasks - Italian-Aligned Models -

		MGSM			MSVAMP		
Model	**Method**	**en**	**It**	**cross**	**en**	**It**	**cross**
	Direct	68.8	60.6	-	73.2	64.8	-
Llamantino2-70	Native-CoT	70.8	61.4	65.0	73.9	66.4	**65.6**
	Native-PAL	**72.0**	**64.6**	**67.2**	**74.3**	**66.2**	70.2
	Direct	64.0	55.2	-	67.9	58.6	-
Llamantino2-7	Native-CoT	66.4	55.6	58.6	68.3	59.4	61.3
	Native-PAL	**68.8**	**58.0**	**60.4**	**70.0**	**61.8**	**63.2**
	Direct	76.0	68.4	-	77.4	69.6	-
Llamantino3-8	Native-CoT	78.2	72.0	73.0	79.2	72.3	74.1
	Native-PAL	**78.8**	**73.2**	**74.6**	**80.3**	**73.3**	**75.6**

Table 14
Accuracies (%) on English and Italian versions of MGSM and MSVAMP using the reasoning methods described in §3.3 (for each model, we reported best performances in **bold**).

16. Results Arithmetic Reasoning Tasks Italian-centred Models

		MGSM			MSVAMP		
Model	**Method**	**en**	**It**	**cross**	**en**	**it**	**cross**
	Direct	62.2	54.6	-	**64.7**	56.3	-
modello-italia	Native-CoT	62.6	55.8	58.4	63.2	57.2	59.7
	Native-PAL	**62.8**	**56.4**	59.2	63.9	**57.8**	60.3
	Direct	44.2	43.6	-	48.6	45.8	-
Minerva-3b	Native-CoT	45.2	43.0	45.2	46.4	45.0	48.7
	Native-PAL	**45.8**	**44.2**	**48.2**	**47.9**	**47.3**	**50.3**
	Direct	42.6	41.8	-	**46.0**	**45.2**	-
Minerva-1b	Native-CoT	41.8	42.0	43.8	45.8	44.6	45.7
	Native-PAL	**43.0**	**42.4**	**45.0**	45.7	45.0	**46.5**
	Direct	46.6	43.0	-	**48.6**	46.0	-
EuroLLM	Native-CoT	46.0	45.8	46.0	46.4	45.4	47.0
	Native-PAL	**47.2**	**47.2**	**48.4**	48.3	**47.0**	**48.5**

Table 15
Accuracies (%) on English and Italian versions of MGSM and MSVAMP using the reasoning methods described in §3.3 (for each model, we reported best performances in **bold**).

17. Results Commonsense, Inference, and Understanding tasks

		XCOPA			XNLI			PAWS-X		
Model	**Method**	**en**	**It**	**cross**	**en**	**It**	**cross**	**en**	**It**	**cross**
GPT-3.5	Direct	93.5	92.6	-	76.2	**67.7**	-	69.5	65.4	-
	Native-CoT	**94.2**	**93.5**	94.1	**77.3**	67.2	71.4	**71.1**	**66.7**	70.3
Llama2-70	Direct	85.6	**80.3**	-	66.3	**56.8**	-	60.4	58.6	-
	Native-CoT	**85.9**	79.6	82.4	68.7	56.2	62.2	**61.5**	**58.9**	60.8
Llama2-7	Direct	**60.8**	**57.8**	-	56.3	**52.2**	-	57.1	**56.0**	-
	Native-CoT	60.6	57.6	58.7	**57.4**	51.9	55.7	**57.8**	55.8	56.6
Llama3-8	Direct	64.3	**61.6**	-	64.8	60.2	-	59.3	**58.4**	-
	Native-CoT	**66.2**	61.1	63.4	**66.3**	**61.3**	65.6	**60.2**	58.2	60.6
Mixtral8x7	Direct	66.2	56.5	-	**47.6**	43.4	-	**59.8**	57.2	-
	Native-CoT	**67.1**	**58.6**	60.4	47.4	**42.9**	45.6	59.3	**57.8**	60.3
Mistral-7	Direct	**62.4**	57.6	-	**43.8**	**41.2**	-	58.0	56.5	-
	Native-CoT	61.6	**58.3**	60.1	43.3	40.7	41.6	**60.4**	**57.3**	59.8
Phi-3	Direct	63.8	62.6	-	63.5	61.2	-	58.9	58.3	-
	Native-CoT	**64.5**	**63.7**	64.1	**65.0**	**63.1**	64.8	**60.7**	**59.8**	60.4
		Italian-aligned								
Llamantino2-70	Direct	84.1	81.6	-	65.1	57.9	-	60.6	60.4	-
	Native-CoT	**85.2**	**82.5**	82.8	**66.3**	**58.6**	61.7	**62.0**	**61.5**	62.4
Llamantino2-7	Direct	60.5	56.3	-	56.0	53.5	-	**56.7**	57.4	-
	Native-CoT	**60.8**	**57.8**	59.6	**56.9**	**54.6**	55.8	56.3	**57.7**	56.9
Llamantino3-8	Direct	63.8	62.7	-	63.4	61.6	-	58.5	59.8	-
	Native-CoT	**64.7**	**64.1**	64.5	**63.9**	**62.7**	64.9	**58.7**	**60.2**	61.4
		Italian-centerd								
modello-italia	Direct	57.6	56.6	-	63.3	**55.7**	-	57.2	55.6	-
	Native-CoT	**58.3**	**56.8**	56.8	**64.3**	56.7	61.6	**58.3**	**57.7**	58.3
Minerva-3b	Direct	59.4	**68.8**	-	56.3	**52.2**	-	57.1	56.0	-
	Native-CoT	**60.6**	67.5	67.3	**56.4**	51.9	55.7	**57.3**	**56.5**	57.2
Minerva-1b	Direct	54.3	**65.3**	-	59.8	49.8	-	49.3	51.4	-
	Native-CoT	**54.8**	64.4	**60.6**	**62.9**	**50.3**	52.3	**50.2**	**52.0**	50.9
EuroLLM	Direct	60.0	59.4	-	56.2	52.4	-	55.8	54.2	-
	Native-CoT	**62.4**	62.1	63.2	**57.0**	**53.6**	56.8	**56.9**	**56.2**	57.1

Table 16
Accuracies (%) on English and Italian versions of XCOPA, XNLI, PAWS-X using the reasoning methods.

Native-CoT

Q: Roger ha 5 palline da tennis. Ha comprato altre 2 lattine di palline da tennis. Ogni lattina contiene 3 palline. Quante palline da tennis ha ora?
A: Roger inizialmente ha 5 palline. 2 lattine con 3 palline ciascuna fanno 6 palline. 5 + 6 = 11. La risposta è 11.
Q: Ci sono nove computer nella sala server. Da lunedì a giovedì, ogni giorno sono stati installati altri cinque computer. Quanti computer ci sono ora nella sala server?
A: Da lunedì a giovedì ci sono 4 giorni. Ogni giorno vengono aggiunti 5 computer. Questo significa che sono stati aggiunti in totale 4 * 5 = 20 computer. Inizialmente ci sono 9 computer, quindi ora ce ne sono 9 + 20 = 29. La risposta è 29.
Q: Sean ha cinque giocattoli. A Natale ha ricevuto due giocattoli da ciascuno dei suoi genitori. Quanti giocattoli ha ora?
A: Ha 5 giocattoli. Ha ricevuto 2 giocattoli da sua madre, quindi dopo ha 5 + 2 = 7 giocattoli. Poi ha ricevuto altri 2 giocattoli da suo padre, quindi in totale ha 7 + 2 = 9 giocattoli. La risposta è 9.

Q: Per fare una tunica servono 2 pezze di tessuto blu e metà di questa quantità di tessuto bianco. Quante pezze di tessuto servono in totale?

GPT-4

Per fare una tunica servono 2 pezze di tessuto blu e metà di questa quantità di tessuto bianco. Prima di tutto, metà di 2 pezze è 1 pezza, quindi serve 1 pezza di tessuto bianco. Sommando il tessuto blu e quello bianco, 2 pezze di blu più 1 pezza di bianco fanno un totale di 3 pezze di tessuto.
Risposta: 3

GPT-3.5

Prima di tutto, la tunica richiede 2 pezze di tessuto blu. Poi, serve la metà di questa quantità di tessuto bianco, cioè 1 pezza di tessuto bianco. Quindi in totale servono 2 + 1 = 3 pezze di tessuto.
Risposta: 3

Mixtral8x7

Per fare una tunica, servono 2 pezze di tessuto blu e la metà di questa quantità di tessuto bianco. La metà della quantità di tessuto bianco sarebbe 2/2 = 1 pezza. Pertanto, per fare una tunica, serviranno in totale 2 (blu) + 1 (bianco) = 3 pezze di tessuto. Risposta: 3

Llama2-7

[:..] Per fare una tunica servono two π blue cotton fabrics and this amount half white cotton fabric. How much fabric do they need in total? A: Per fare una tunica, servono 2 pi (π) di tessuto di cotone blu e metà di questo è uguale a 1 pi (π) di tessuto di cotone bianco. Quindi, in totale, servono 2 * π + (1/2) * π = 3π tessuti di cotone bianco.

Mistral-7

Per fare un abito, servono 2 pezze di tessuto blu e metà di questa quantità di tessuto bianco. Pertanto, servono 2 (blu) + 2 (metà di quanto 2 è per il bianco) = 3.5. Poiché non possiamo avere mezza pezza di tessuto, supponiamo che serva un'ulteriore 0.5 pezze per coprire questa esigenza. Quindi, servono in totale 3.5 x 2 = 7 pezze di tessuto per creare un abito.
La risposta è 7.

Table 17
Example output answers generated by different LLMs prompted via Native-CoT (Section 3.3). In general, LLMs produce good answers, while Smaller LLMs (for example, see Llama2-7) produce misleading answers.