Does the *Order* Matter? Curriculum Learning over Languages

Leonardo Ranaldi [(†)], **Giulia Pucci** [(⋆)], **Andrè Freitas**[(†,∗)]
(†) Idiap Research Institute, Martigny, Switzerland
(∗)Department of Computer Science, University of Manchester, UK
(⋆)Department of Computing Science, University of Aberdeen, UK
{name.surname}@idiap.ch

Abstract

Curriculum Learning (CL) has been emerged as an effective technique for improving the performances and reducing the cost of pre-training Large Language Models (LLMs). The efficacy of CL demonstrated in different scenarios is in the training LLMs by organizing examples from the simplest to the most complex. Although improvements have been shown extensively, this approach was used for pre-training, leaving novel fine-tuning approaches such as instruction-tuning unexplored. In this paper, we propose a novel complexity measure to empower the instruction-tuning method using the CL paradigm. To complement previous works, we propose cognitively motivated measures to determine the complexity of training demonstrations used in the instruction-tuning paradigm. Hence, we experiment with the proposed heuristics first in English and then in other languages. The downstream results show that delivering training examples by complexity ranking is also effective for instruction tuning, as it improves downstream performance while reducing costs. Furthermore, the technique can be easily transferred to languages other than English, e.g., Italian and French, without any adaptation, maintaining functionality and effectiveness.

Keywords: Instruction-tuning, Multi-lingual efficient tuning, Curriculum Learning

1. Introduction

The evolution of the Large Language Models (LLMs) ecosystem is intrinsically related to the development of effective refinement methods that promote access and improve empathy from mainstream audiences. The introduction of cutting-edge techniques involving humans in refinement processes (Ouyang et al., 2022; Rafailov et al., 2023) attracts attention due to its outstanding effectiveness and versatility. The keystone lies in the powers of LLMs to grasp and act upon human instructions, where this alignment is attributed to the additional tuning process (Gupta et al., 2022; Wei et al., 2022). This paradigm is giving rise to numerous studies proposing instruction-tuning methods to elicit models to follow more complex instructions, improving performance in various tasks (Honovich et al., 2023).

Ranaldi and Freitas (2024) demonstrated that producing demonstrations that deliver step-by-step reasoning improves instruction-tuning performance and stimulates LLMs' reasoning ability. Wang et al. (2023); Zhou et al. (2023) observed significant benefits related to the quantity and quality of instruction data that Chen et al. (2024); Muennighoff et al. (2023); Ranaldi et al. (2023a); Tanwar et al. (2023) transferred in multi-lingual scenarios. Although earlier works have offered important insights for maximizing the effective operation of the instruction-tuning paradigm, these focus on engineering demonstrations by naïvely leaving for training using batches of demonstrations randomly sampled from training corpora.

Since the emergent refinement techniques aim to emulate human-like cognitive learning processes, the incremental organization training examples, known as Curriculum Learning (CL) (Bengio et al., 2009), could constitute a logically coherent and methodologically robust learning strategy for instruction-tuned language models. Several works have leveraged CL in pre-training (Nagatsuka et al., 2021; Cui et al., 2022) and fine-tuning (Zhou et al., 2020; Xu et al., 2020; Spitkovsky et al., 2010; Zhang et al., 2021) phases, proposing complexity measures leveraging the structure of the language (Ranaldi et al., 2023b) to achieve better performance and computational efficiency results. However, the nature of demonstrations underlying the instruction-tuning technique makes applying complexity metrics proposed by previous works challenging.

In this paper, in order to bring the instruction-tuning method to the human learning process, we propose a complexity measure to deliver training demonstrations in a logically motivated manner. By getting inspiration from the Curriculum Learning approach, we propose an instruction-tuning methodology starting from simpler demonstrations and gradually increasing complexity. Besides previous works, we aim to emulate human learning by quantifying the cognitive abilities required to solve problems because since text structure enough is limiting as a heuristic measure of complexity. Therefore, during the instruction-tuning phase, we deliver the demonstrations following Bloom's taxonomy (Adams, 2015), as shown in Figure 1.

LREC-COLING 2024, pages 5212–5220

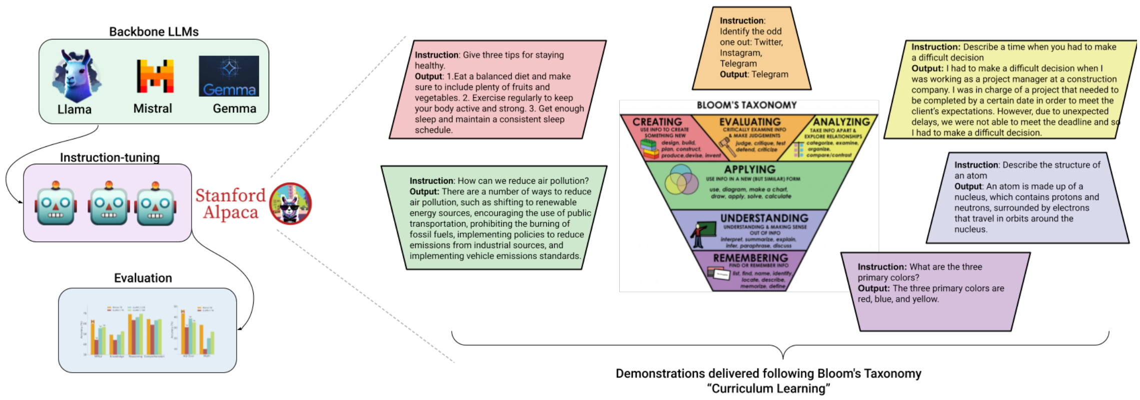

Figure 1: Our Curriculum Learning heuristic based on Bloom's Taxonomy. In particular, the basic pipeline for instruction tuning is shown on the left, and the right is the policy, which is the strategic part of our work.

To observe the effects of cognitively motivated instruction-oriented supervised fine-tuning (instruction-tuning), we employ Llama2-7b (Touvron et al., 2023) as our baseline model, and Alpaca (Li et al., 2023) as the demonstration corpus. Hence, we conduct the baseline instruction-tuning as proposed in (Li et al., 2023) by providing demonstrations without accounting for their sequence and adhering to our human-inspired approach. We evaluate the functionality of our approach using tasks involving both mathematical reasoning Multilingual Grade School Match (MGSM) and natural language understanding MultiLingual Question Answering (MLQA). Furthermore, we apply the same pipeline to additional languages to investigate if our approach could be transferred to them, adapting two multi-task benchmarks to the specific settings. The final results show that cognitively motivated instruction-tuning brings benefits in English and additional languages by improving LLMs' abilities to solve different types of tasks.

2. Method

Instruction-tuning is critical to Large Language Models (LLMs) for achieving better instruction following and task adaptation capabilities. Although previous works studied the impacts on the downstream performances related to data quality from a human-like perspective, they left the training phase unexplored. Following the Curriculum Learning (CL) strategy, where training algorithms can achieve better results when training data are presented according to the model's current skills (Bengio et al., 2009), we propose an additional pre-tuning phase, as shown in Figure 1. In particular, using the original instruction-tuning approach described in Section 2.1, we introduce an annotation phase that estimates the complexities of demonstrations used during the instruction-tuning via cognitively motivated heuristics introduced in Section 2.2.

2.1. The Instruction-tuning Paradigm

Ouyang et al. (2022); Wei et al. (2022) fine-tuned LLMs using the instruction-tuning method based on demonstrations, which are instruction-response corpora, to make LLMs more scalable and improve zero-shot performance. In this way, the LLMs backbone are fed with a set of demonstrations structured as (i, x, y), where i is an instruction describing the task's requirements, x is the input, which can be optional, and y is the output for the given task. The goal of this method is to minimize the function $f(y)$:

$$f(y) = \arg\min_{\theta} \log p_{\theta}(y \mid i, x) \tag{1}$$

where θ are model learnable parameters.

Many studies have shown the elasticity of this paradigm by proposing customized instruction in multi and cross-lingual settings (Ranaldi et al., 2023a; Ranaldi and Pucci, 2023a; Chen et al., 2024). However, in this work, we use the original Alpaca (Li et al., 2023) that is synthetic-generated instructions in English. The demonstrations cover different tasks, which can be grouped by category as reported in Figure 2.

2.2. Curriculum Learning

Since the instruction-tuning demonstrations aim to instruct LLMs to solve general tasks by following instructions emulating human learning, delivering examples in order of complexity can improve performance. Curriculum Learning (CL) (Bengio et al., 2009) is a training method based on the idea that training algorithms can achieve better results when training data are presented in accordance with the model's current abilities. Although CL-based solutions have shown effective improvements in pre-training and fine-tuning time, using the structure as a complexity metric is definitely limited for the purpose of this paradigm. Hence, we propose a logically motivated metric leveraging Bloom's taxonomy (Adams, 2015) as the connection metric.

Complexity Metric Bloom's taxonomy is a cognitive psychology instrument that classifies educational objectives. This taxonomy identifies six levels of cognitive learning, from the simplest to the most complex: *remembering*, *understanding*, *applying*, *analyzing*, *evaluating*, and *creating*. By construction, it can be a strategic measure for bringing the instruction-tuning method closer to the human learning process by quantifying the complexity of demonstrations by taking a human-like perspective.

Annotation Prompt

```
Given the following task described
in the triple Instruction, Input,
Output.
##Instruction:  Given two words,
think of a sentence that is re-
lated to both words.
##Input:  "Title and Dream"
##Output:  "Dream of a title."
Choose one of the following abili-
ties:
-remember
-understand
-apply
-analyse
-evaluate
-create
Answer:[ability]
```

Table 1: Our prompting approach for choosing Bloom's taxonomy level.

Applying Complexity Heuristics Using Bloom's taxonomy, we systematically estimate the complexity of the demonstrations by assigning them to one of the six abilities mentioned previously. In order to produce a robust evaluation, we systematically prompt `GPT-3.5-turbo` using the prompt defined in Table 1. Then, behind assigning each demonstration its cognitive level, we reorder the demonstrations of the same level by length, that is, by the number of tokens present. Finally, we perform instruction-tuning as described in Section 2.1 by delivering the demonstrations during the tuning phase according to the proposed heuristics.

3. Experimental Setup

In order to assess the performance of the complexity measures proposed in Section 2, we introduce several benchmarks (Section 3.1) on which we applied systematic tuning (Section 3.2) and evaluation (Section 3.3) pipelines.

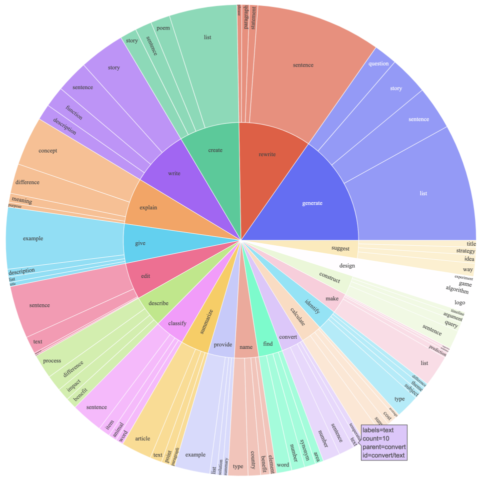

Figure 2: Typology of the demonstrations in the Stanford Alpaca dataset. Illustration from (Santilli and Rodolà, 2023). In our work, we have dealt eclectically with the topology of abilities in the inner loop by restricting them to only the six proposed by Bloom (Adams, 2015).

3.1. Benchmarks

In this work, it is proposed a comprehensive evaluation of different languages, in particular are used two multilingual (MGSM (Shi et al., 2022), MLQA (Lewis et al., 2020)) and two multi-task (MMLU (Hendrycks et al., 2021) and BBH (Suzgun et al., 2022)) benchmarks. MGSM and MLQA focus on mathematical reasoning and understanding questions and answers in different languages. MMLU and BBH, being multi-task benchmarks, include subtasks related to Boolean expressions and QA on basic-level subjects (e.g., chemistry, physics). However, we decided to introduce them to observe whether our approach degrades performance in these tasks. The first two datasets selected are appropriately constructed for multi-language testing, while the second two are available only in English. Hence, we did a preliminary translation step as outlined below.

Multilingual Grade School Match (MGSM) (Shi et al., 2022) evaluates the problem-solving abilities in multilingual scenarios. The original version, well known as GSM8K, is composed of English problems. Each example has the following structure: a mathematical problem in natural language and a target answer in Arabic number. Shi et al. (2022), in their contribution, i.e., MGSM, selected the first 250 examples from the official list of examples in GSM8K and translated them manually into 11 different languages, maintaining the structure of the input and output.

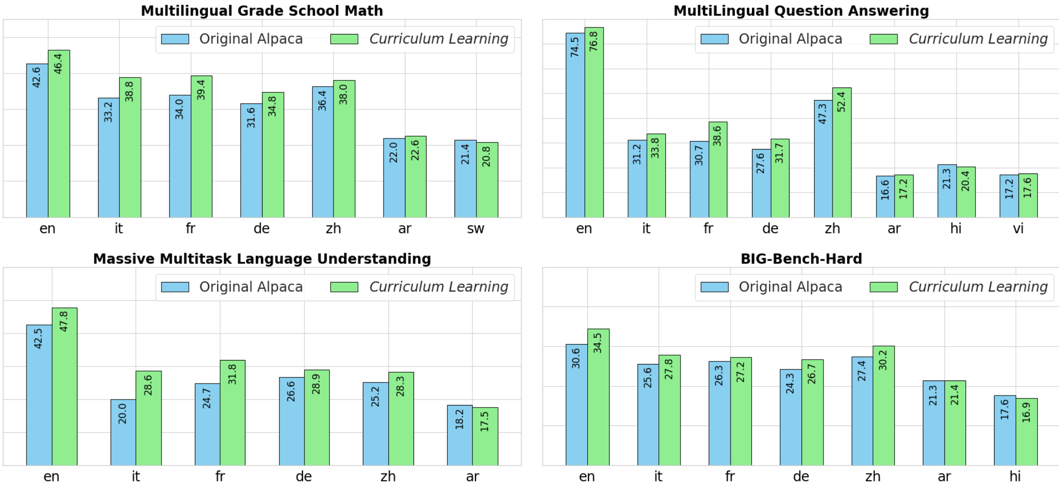

Figure 3: Accuracies (%) on benchmarks presented in Section 3.1 using original Alpaca (Li et al., 2023) pipeline and our *Curriculum Learning* pipeline introduced introduced in Section 2.2.

MultiLingual Question Answering (MLQA) (Lewis et al., 2020) evaluates multilingual question answering performance. The benchmark comprises over 5K extractive QA instances in several languages in the SQuAD (Rajpurkar et al., 2016) format. MLQA is highly parallel, with QA instances aligned across four languages on average. Although comprising different languages, some languages, such as Italian, are not represented. To conduct the experiments uniformly, we have translated the examples as also done in the forthcoming MMLU and BBH.

Massive Multitask Language Understanding (MMLU) (Hendrycks et al., 2021) measures knowledge of the world and problem-solving problems in multiple subjects with 57 subjects across STEM, humanities, social sciences, and other areas. The benchmark is native in English; however, we translated it into five additional languages[1].

BIG-Bench Hard (BBH) (Suzgun et al., 2022) is a subset of challenging tasks related to navigation, logical deduction, and fallacy detection. Again, the benchmark is native English, and we have translated it into five languages[1].

3.2. Models Instruction-tuning

All models are tuned following the official Alpaca repository. The translated versions available (open-source Alpaca) have been used for each specific language. We used the alpaca_LoRA (Hu et al., 2021) code, adopting the same hyperparameters to align the results with the state-of-the-art models. We performed the fine-tuning with a single epoch and a batch-size of 128 examples, running our experiments on a workstation equipped with two Nvidia RTX A6000 with 48 GB of VRAM.

3.3. Evaluation

We then divide the evaluation criteria into two parts: 1) MGSM and MLQA are evaluated using a zero-shot prompting approach and estimating accuracy by measuring exact match values in the zero-shot setting; 2) MMLU and BBH are evaluated using the open-source framework InstructEval[2]. For each model, the parts of benchmarks related to the specific language are used (e.g., for zh that is zh-Alpaca data from MLQA, XQUAD, MMLU, and BBH in Chinese are used).

4. Results

The instruction-tuning process inspired by cognitive learning brings consistent benefits, as shown in Figure 3. In particular, as shown in Table 2, the models tuned following the complexity heuristics proposed in Section 2 outperform the original settings by 2.2 points on average. However, as discussed in Section 4.1, there is an average difference between the languages. Furthermore, the proposed method shows sensible improvements as the demonstrations decrease, as described in Section 4.2.

Finally, cognitively motivated instruction-tuning benefits further open-source Large Language Models (LLMs). In fact, as discussed in Section 4.3, scaling the pipeline on further models reveals that the order affects the final performance.

[1]We performed translations using the Google translator API from English to Chinese (zh), Italian (it), Arabic (ar), Spanish (es), German (de). Resources available here

[2]https://github.com/declare-lab/instruct-eval

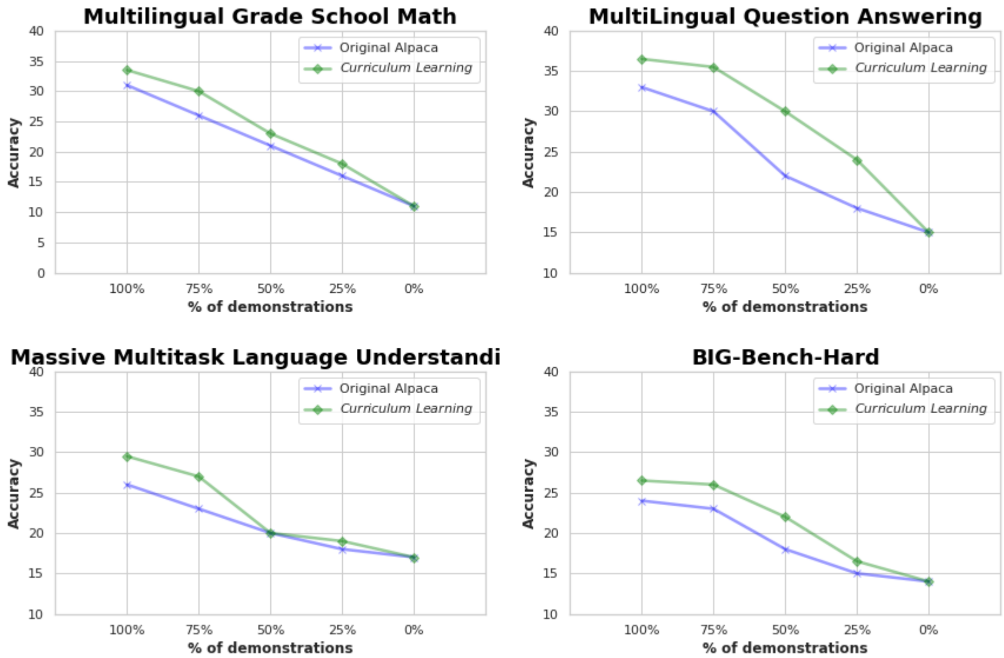

Figure 4: Evaluation of proposed benchmarks using standard Alpaca-like settings and ordering demonstrations using the heuristics proposed in Section 2. In contrast to the experiment proposed in Figure 3, here we systematically describe the demonstrations used to perform instruction-tuning.

4.1. The Language Matter

Although the LLMs refined via cognitively motivated order demonstrations have more significant results than the baselines, the proposed method has limitations. In fact, as shown in Figure 3, not all languages benefit from this method; in particular, low-resource languages such as Hindi and Swahili seem to achieve the same results. On the other side of the coin, high-resource languages such as English, Italian and Chinese seem to have robust benefits. We estimate languages using CommonCrawl (Common Crawl, 2021) as a benchmark, as shown in Table 3.

However, although the method we proposed achieved poor results in low-resource languages, the starting baselines are very low. Therefore, in Section 4.2, to observe the impact of the type of demonstrations from a macroscopic point of view, we study whether decreasing the number of demonstrations equally provided following our order produces the desired effects.

4.2. The Power of the Demonstrations

Curriculum-based instruction-tuning is more efficient as the number of demonstrations decreases. Figure 4 shows the average performance of the models evaluated on the benchmarks introduced in Section 3. In particular, it can be observed that in both the MGSM arithmetic task and the MLQA understanding task, models instructed with cognitively motivated orders outperform models instructed with randomly provided demonstrations. This result confirms that the proposed method does indeed work, as although fewer demonstrations are present, they

Task	avg Alpaca	avg Curriculum	δ
MGSM	31.5	32.8	+1.3
MLQA	33.0	35.6	+2.6
MMLU	26.1	29.4	+3.3
BBH	24.7	26.4	+1.7

Table 2: Averages of the results on proposed benchmarks. The column δ indicates the difference between avg-Curriculum and avg-Alpaca in custom language Learning (Alpaca).

Language	Percentage
English (en)	46.3%
Russian (ru)	6.0%
German (de)	5.4%
Chinese (zh)	5.3%
French (fr)	4.4%
Japanese (ja)	4.3%
Spanish (es)	4.2%
Italian (it)	3.9%
Other	19.1%

Table 3: Language distribution of CommonCrawl (Common Crawl, 2021).

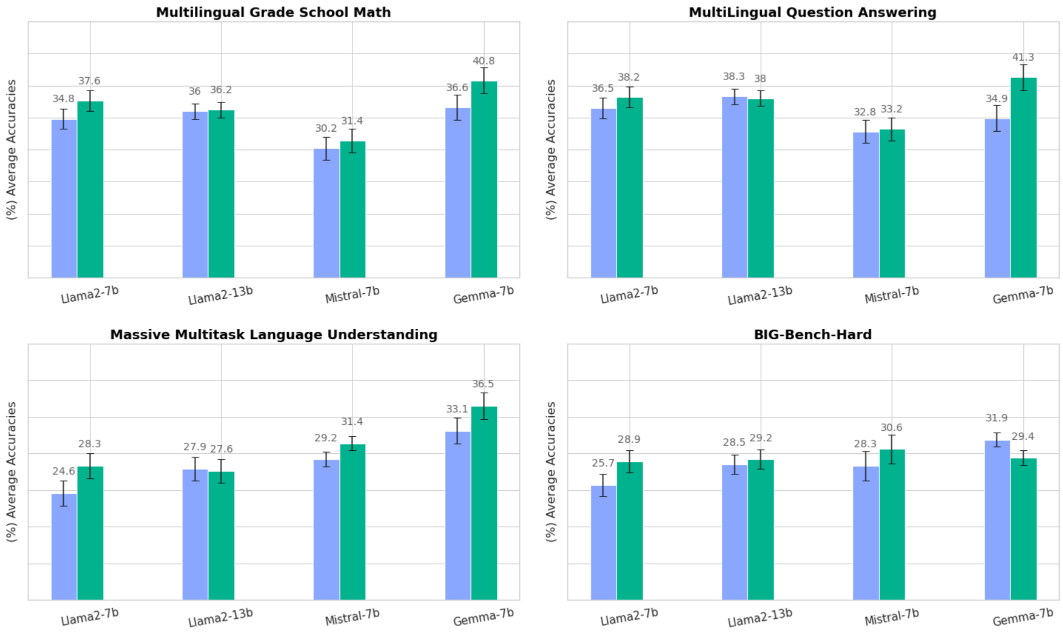

Figure 5: Average accuracies (%) on benchmarks presented in Section 3.1 using additional Large Language Models instruction-tuned on original Alpaca (Li et al., 2023) pipeline and our *Curriculum Learning* pipeline introduced introduced in Section 2.2.

make the models learn better if they are ordered.

Although these results appear to be stable for Llama-7b, the experiments are not complete. In Section 4.3, we propose the same experimental pipeline by introducing additional LLMs from different families and then training them in different ways.

4.3. Scaling Curriculum Learning to other Models

Learning heuristics inspired by cognitive mechanisms are easily scalable to different LLMs. Figure 5 shows the accuracies obtained from further commonly trained models using Alpaca and customized versions for different languages.

In particular, we select Llama2-7b, Llama2-13b (Touvron et al., 2023), Gemma-7b (Team et al., 2024)[3], and Mistral-7b (Jiang et al., 2023). The choice was mainly dictated by the common use with which the models were instructed to follow the instructions using ALpaca and the language-specific derivatives. As can be seen from Figure 5, the model that benefits the most is Gemma-7b, while on Mistral-7b, there seems to be less effect. Furthermore, comparing Llama2-7b and Llama2-13b from the same family but with different numbers of parameters, it can be observed that the model with more parameters benefits less from this technique. We assume this is due firstly to the higher primary performance and secondly to the relationship between the number of parameters and the pretty poor data set. In future studies, we will continue to investigate the strategic impact of the quality and quantity of instructions that LLMs need to optimize their instruction-tuning phases.

[3]Note that we have added Gemma-7b (it is supervised fine-tuned as Alpaca-like manner) to our evaluation in the camera-ready version.

5. Limitations & Future Works

The cognitively motivated metrics used to provide examples during instruction-tuning, as proposed in Section 2, have shown multiple benefits on the benchmarks introduced in Section 3. Detailed analyses have been extensively discussed in Section 4, touching on strengths and weaknesses. Among the strengths are the versatility and scalability of the approach across different models. On the other hand, there needs to be more effectiveness in low-resource languages and models with many parameters. In future developments, we intend to improve this aspect by considering the introduction of structured ecosystems (Zanzotto et al., 2020; Ranaldi and Pucci, 2023b) and multi- and cross-lingual approaches. Finally, we would like to investigate the impact of previously seen demonstrations during in-training and data contamination (Ranaldi et al., 2023c, 2024), as well as the behaviors that Large Language Models exhibit in interaction with users (Ranaldi and Pucci, 2023c).

6. Conclusion

In this work, inspired by Curriculum Learning, we proposed a cognitively motivated instruction-tuning technique. Using Bloom's taxonomy as a complexity metric, we organized instruction-tuning corpora in different languages, which we then used to season the instruction-tuning phase. In order to produce a robust evaluation, we tested different models in various languages. From the final results, we observed that this technique brings significant benefits in reasoning tasks and question answering. Through this study, we aim to narrow the gap between Large Language Models and instruction inspired by human cognitive processes hoping that this research-line could continue in this direction.

Acknowledgements

This paper has been partially supported by the SINFONIA project funded by Innosuisse n. 104.170 IP-ICT. We would like to thank the reviewers for their constructive comments.

Bibliographical References

Nancy E Adams. 2015. Bloom's taxonomy of cognitive learning objectives. *Journal of the Medical Library Association: JMLA*, 103(3):152–153.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48.

Pinzhen Chen, Shaoxiong Ji, Nikolay Bogoychev, Andrey Kutuzov, Barry Haddow, and Kenneth Heafield. 2024. Monolingual or multilingual instruction tuning: Which makes a better alpaca. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1347–1356, St. Julian's, Malta. Association for Computational Linguistics.

Common Crawl. 2021. Common crawl 2021. Web. Accessed: 2023-12-12.

Yiming Cui, Wanxiang Che, Shijin Wang, and Ting Liu. 2022. Lert: A linguistically-motivated pre-trained language model.

Prakhar Gupta, Cathy Jiao, Yi-Ting Yeh, Shikib Mehri, Maxine Eskenazi, and Jeffrey Bigham. 2022. InstructDial: Improving zero and few-shot generalization in dialogue through instruction tuning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 505–525, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding.

Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. 2023. Unnatural instructions: Tuning language models with (almost) no human labor. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14409–14428, Toronto, Canada. Association for Computational Linguistics.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2020. MLQA: Evaluating cross-lingual extractive question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7315–7330, Online. Association for Computational Linguistics.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Alpacaeval: An automatic evaluator of instruction-following models. https://github.com/tatsu-lab/alpaca_eval.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2023. Crosslingual generalization through multitask finetuning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15991–16111, Toronto, Canada. Association for Computational Linguistics.

Koichi Nagatsuka, Clifford Broni-Bediako, and Masayasu Atsumi. 2021. Pre-training a BERT with curriculum learning by increasing block-size of input text. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 989–996, Held Online. INCOMA Ltd.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Federico Ranaldi, Elena Sofia Ruzzetti, Dario Onorati, Leonardo Ranaldi, Cristina Giannone, Andrea Favalli, Raniero Romagnoli, and Fabio Massimo Zanzotto. 2024. Investigating the impact of data contamination of large language models in text-to-sql translation.

Leonardo Ranaldi and Andre Freitas. 2024. Aligning large and small language models via chain-of-thought reasoning. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1812–1827, St. Julian's, Malta. Association for Computational Linguistics.

Leonardo Ranaldi and Giulia Pucci. 2023a. Does the English matter? elicit cross-lingual abilities of large language models. In *Proceedings of the 3rd Workshop on Multi-lingual Representation Learning (MRL)*, pages 173–183, Singapore. Association for Computational Linguistics.

Leonardo Ranaldi and Giulia Pucci. 2023b. Knowing knowledge: Epistemological study of knowledge in transformers. *Applied Sciences*, 13(2).

Leonardo Ranaldi and Giulia Pucci. 2023c. When large language models contradict humans? large language models' sycophantic behaviour.

Leonardo Ranaldi, Giulia Pucci, and Andre Freitas. 2023a. Empowering cross-lingual abilities of instruction-tuned large language models by translation-following demonstrations.

Leonardo Ranaldi, Giulia Pucci, and Fabio Massimo Zanzotto. 2023b. Modeling easiness for training transformers with curriculum learning. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 937–948, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Leonardo Ranaldi, Elena Sofia Ruzzetti, and Fabio Massimo Zanzotto. 2023c. PreCog: Exploring the relation between memorization and performance in pre-trained language models. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 961–967, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Andrea Santilli and Emanuele Rodolà. 2023. Camoscio: an italian instruction-tuned llama.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. 2022. Language models are multilingual chain-of-thought reasoners.

Valentin I. Spitkovsky, Hiyan Alshawi, and Daniel Jurafsky. 2010. From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 751–759, Los Angeles, California. Association for Computational Linguistics.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. 2022. Challenging big-bench tasks and whether chain-of-thought can solve them.

Eshaan Tanwar, Subhabrata Dutta, Manish Borthakur, and Tanmoy Chakraborty. 2023. Multilingual llms are better cross-lingual in-context learners with alignment.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le

Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Christian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, Justin Mao-Jones, Katherine Lee, Kathy Yu, Katie Millican, Lars Lowe Sjoesund, Lisa Lee, Lucas Dixon, Machel Reid, Maciej Mikuła, Mateo Wirth, Michael Sharman, Nikolai Chinaev, Nithum Thain, Olivier Bachem, Oscar Chang, Oscar Wahltinez, Paige Bailey, Paul Michel, Petko Yotov, Pier Giuseppe Sessa, Rahma Chaabouni, Ramona Comanescu, Reena Jana, Rohan Anil, Ross McIlroy, Ruibo Liu, Ryan Mullins, Samuel L Smith, Sebastian Borgeaud, Sertan Girgin, Sholto Douglas, Shree Pandya, Siamak Shakeri, Soham De, Ted Klimenko, Tom Hennigan, Vlad Feinberg, Wojciech Stokowiec, Yu hui Chen, Zafarali Ahmed, Zhitao Gong, Tris Warkentin, Ludovic Peran, Minh Giang, Clément Farabet, Oriol Vinyals, Jeff Dean, Koray Kavukcuoglu, Demis Hassabis, Zoubin Ghahramani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah Fiedel, Evan Senter, Alek Andreev, and Kathleen Kenealy. 2024. Gemma: Open models based on gemini research and technology.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Raghavi Chandu, David Wadden, Kelsey MacMillan, Noah A. Smith, Iz Beltagy, and Hannaneh Hajishirzi. 2023. How far can camels go? exploring the state of instruction tuning on open resources.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners.

Benfeng Xu, L. Zhang, Zhendong Mao, Quan Wang, Hongtao Xie, and Yongdong Zhang. 2020. Curriculum learning for natural language understanding. In *Annual Meeting of the Association for Computational Linguistics*.

Fabio Massimo Zanzotto, Andrea Santilli, Leonardo Ranaldi, Dario Onorati, Pierfrancesco Tommasino, and Francesca Fallucchi. 2020. KERMIT: Complementing transformer architectures with encoders of explicit syntactic interpretations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 256–267, Online. Association for Computational Linguistics.

Mingliang Zhang, Fandong Meng, Yunhai Tong, and Jie Zhou. 2021. Competence-based curriculum learning for multilingual machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2481–2493, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. Lima: Less is more for alignment.

Yikai Zhou, Baosong Yang, Derek F. Wong, Yu Wan, and Lidia S. Chao. 2020. Uncertainty-aware curriculum learning for neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6934–6944, Online. Association for Computational Linguistics.