ISSN 2499-4553

IJCoL

Italian Journal of Computational Linguistics

Rivista Italiana di Linguistica Computazionale

Volume 10, Number 1
june 2024

aA Accademia university press

Registrazione presso il Tribunale di Trento n. 14/16 del 6 luglio 2016

Rivista Semestrale dell'Associazione Italiana di Linguistica Computazionale (AILC)

Associazione Italiana di Linguistica Computazionale

direttore responsabile
Michele Arnese

isbn 9791255000983

Accademia University Press
via Carlo Alberto 55
I-10123 Torino
info@aAccademia.it
www.aAccademia.it/IJCoL_10_1

aAccademia university press

Accademia University Press è un marchio registrato di proprietà
di LEXIS Compagnia Editoriale in Torino srl

IJCoL

Volume 10, Number 1
june 2024

CONTENTS

Adapting BLOOM to a new language: A case study for the Italian

Pierpaolo Basile*
Università di Bari Aldo Moro

Lucia Siciliani**
Università di Bari Aldo Moro

Elio Musacchio†
Università di Bari Aldo Moro

Marco Polignano‡
Università di Bari Aldo Moro

Giovanni Semeraro§
Università di Bari Aldo Moro

The BLOOM Large Language Model is a cuttingI think that the authors' way of doing self-training is quite interesting. I suggest writing about self-training in the abstract and generally expose self-training as one of the paper's contributions.-edge open linguistic model developed to provide computers with natural language understanding skills. Despite its remarkable capabilities in understanding natural language by capturing intricate contextual relationships, the BLOOM model exhibits a notable limitation concerning the number of included languages. In fact, Italian is not included among the languages supported by the model, making its use challenging in this context. Within this study, we explore the language adaptation strategy based on continuing training on language-specific data. Moreover, we fine-tune both the BLOOM and the adapted models on several instruction datasets and different downstream classification tasks over EVALITA datasets. It has been observed that language adaptation followed by instruction-based fine-tuning is shown to be effective in correctly addressing a task never seen by the model in a new language learned on language-specific data.

1. Introduction

As language diversity becomes increasingly important in the digital age, the capability of a Natural Language Understanding model to handle a wide array of languages gains significance. Large Language Models (LLMs) have emerged as excellent approaches for comprehending, generating, and manipulating human language with unprecedented accuracy and fluency (Naveed et al. 2023).

They can grasp nuances, idioms, and even ambiguous phrases, enabling more accurate sentiment analysis, question answering, and information retrieval tasks. This enhanced understanding contributes to more effective communication between humans and machines, fostering seamless interactions across various applications. LLMs pos-

* Dept. of Computer Science - Via E. Orabona n.4, 70125 Bari, Italy.
E-mail: pierpaolo.basile@uniba.it
** Dept. of Computer Science - Via E. Orabona n.4, 70125 Bari, Italy. E-mail: lucia.siciliani@uniba.it
† Dept. of Computer Science - Via E. Orabona n.4, 70125 Bari, Italy. E-mail: elio.musacchio@uniba.it
‡ Dept. of Computer Science - Via E. Orabona n.4, 70125 Bari, Italy. E-mail: marco.polignano@uniba.it
§ Dept. of Computer Science - Via E. Orabona n.4, 70125 Bari, Italy.
E-mail: giovanni.semeraro@uniba.it

sess remarkable generalization capabilities, allowing them to perform well on tasks they were not explicitly trained for, also in a multilingual fashion. Among the largest and most effective Large Language Models can be found BLOOM (Scao et al. 2022), a 176B-parameter open-access language model designed and built thanks to the collaboration of hundreds of researchers. BLOOM is a decoder-only Transformer language model that was trained on a large corpus comprising hundreds of sources in 46 natural and 13 programming languages, culminating in a comprehensive dataset that spans 59 languages in total. Nevertheless, it excludes some of the world's most widely spoken languages, including Russian, Korean, and Italian, raising the need for a more inclusive linguistic approach. Training an effective LLM focused solely on a particular language is a prohibitive challenge, given the substantial volumes of data and resources required for such a task. At the same time, tackling downstream tasks in a specific language effectively necessitates a model with a comprehensive understanding of that language.

Our hypothesis focuses on the language adaptation methodology, which is particularly fascinating for addressing the challenge of transferring knowledge from a pre-trained Language Model (LM) to a specific application language. In this context, we aim to adapt BLOOM models to work with a new language, such as Italian, using language-specific data.

Indeed, we evaluated the adapted models after a phase of fine-tuning on several instruction datasets and different classification tasks using Italian data. Our experiments demonstrate that the language adaptation process improves the ability of the model if executed for the same language of the evaluating data as already proved in our previous work (Basile et al. 2023a). One of the most important aims of our work is to execute adaptation and fine-tuning on limited computational resources; for that reason, we adopt Selective Parameter Training. This strategy entails training solely a portion of the pre-trained model's parameters using language-specific data. The model can be customised through selective adjustment of specific parameters to enhance its adaptation to the target language while capitalizing on the general knowledge acquired during pre-training. These techniques are commonly denoted as Parameter-Efficient Finetuning Techniques (PEFT) (Hu et al. 2023). Moreover, we consider the smallest BLOOM models of 1.7 billion parameters in order to fit the training process on a single affordable GPU[1]. We want to prove that fostering innovation and building effective LLMs is possible only by using open resources. The paper is structured as follows: an overview of language adaptation strategies is reported in Section 2, while our adaptation and fine-tuning pipeline is described in Section 3. In Section 4, we present a thorough evaluation and discussion of results. Finally, Section 5 closes the paper, reporting the conclusion and future work.

2. Language Adaptation Approaches

LLMs, such as GPT (Brown et al. 2020), Vicuna (Chiang et al. 2023), LLaMA (Touvron et al. 2023), or BLOOM (Scao et al. 2022), are trained on vast amounts of text data from diverse sources, which gives them a broad understanding of language and context. Nonetheless, it is important to note that the general knowledge inherent in these models might not be optimised for a particular language (Nowakowski et al. 2023). For this reason, language adaptation can strongly support the model's capacity to navigate and address downstream tasks in a specific language effectively. Language adaptation of

1 All our training and evaluation steps are conducted on a single NVIDIA RTX A6000 with 48GB of RAM.

LLMs refers to the process of tuning a pre-trained LM to work effectively with a specific target language. In the scientific literature, different approaches for language adaptation have been recently proposed (Yong et al. 2023). Among them, we can distinguish i) continuing the pre-training on new data (Chau, Lin, and Smith 2020), ii) creating a model adapter (Wang et al. 2021), iii) training a random subset of the model parameters (Ansell et al. 2022). Furthermore, some works also extend these approaches by refining the vocabulary that is learned by the model and its tokenizer. This process is called *vocabulary augmentation*, where the idea is to incorporate tokens that better fit the required language. In this strategy, first, the vocabulary is modified, some works use a language mapping-based technique (Wang et al. 2019) or an Entropy-based approach for low-resource languages tasks (Nag et al. 2023), and then the model is further trained to adapt to the new modified vocabulary.

In this work, we focus mainly on continuing the pre-training on new language-specific data without augmentation of the vocabulary since it is the easiest strategy to implement and is well supported by the more recent software library. One of the main drawbacks of this approach is that the resulting model retains the same number of parameters as the original one. To overcome this limitation, we adopted a **Parameter-Efficient Fine-Tuning (PEFT)** strategy based on LoRA during the training and released for each new model only the adapter that contains only a portion of the parameters of the original model. PEFT techniques (Hu et al. 2023) have emerged as a valuable strategy for streamlining the adaptation of pre-trained language models (PLMs) across various downstream applications. PEFT methods tackle this challenge by selectively fine-tuning only a small subset of additional model parameters. Consequently, the computational and storage costs associated with PEFT are notably diminished. Noteworthy recent advancements in PEFT have showcased remarkable performance comparable to that achieved through complete fine-tuning. This highlights the effectiveness of PEFT methods in achieving a balance between computational efficiency and maintaining competitive model performance.

Among PEFT methodologies, **LoRA** (Hu et al. 2022) reduces the number of trainable parameters within a neural network. LoRA is a mathematically rigorous approach that delves into the concept of the *intrinsic dimension* of weight matrices in pre-trained neural networks. Unlike conventional weight matrices that exhibit full rank, where each weight is distinct and cannot be expressed as a combination of others, LoRA unveils an intriguing phenomenon. The weights demonstrate a lower intrinsic dimension when pre-trained language models are fine-tuned for new tasks. This suggests that the weights can be represented in a smaller matrix or possess a lower rank. This mathematical discovery carries profound implications. In LoRA, the weight update matrix displays a diminished rank during the backpropagation process. This phenomenon can be attributed to the pre-training phase already capturing substantial information, thereby allowing the fine-tuning stage to primarily concentrate on task-specific adjustments. In essence, LoRA presents a compelling strategy for parameter reduction by harnessing the concept of intrinsic dimensionality within weight matrices. In LoRA, a critical step involves fully loading the model into the utilised Graphics Processing Unit (GPU) memory. For that reason, we consider only the smallest model of the BLOOM family (1.7B) without investigating methods for reducing model size as Model Distillation (Jiao et al. 2020) and Quantization (Guo 2018).

Through fine-tuning based on PEFT and LoRA, we can continue training the original model on new language-specific data. In this work, we build an Italian corpus based on the March 2024 dump of the Italian versions of Wikipedia, Wikinews, and Wikibooks.

3. Adaptation Pipeline

Starting from BLOOM-1b7[2], we build an adapted Italian model called BLOOM-IT-1b7 obtained by fine-tuning the original BLOOM models on the Italian corpus.

Moreover, we fine-tune both the original and the adapted model on several instruction datasets and EVALITA tasks. In particular, we rely on the following datasets:

- Camoscio[3] is an Italian translation with ChatGPT of the Stanford Alpaca (Touvron et al. 2023; Wang et al. 2023) dataset. The Camoscio dataset was used to build an Italian instruction-tuned version of LLaMA (Santilli and Rodolà 2023).
- Dolly-IT[4] (Basile et al. 2023a) is an Italian translation of the Dolly Instruction Dataset (Conover et al. 2023). Dolly[5] is made of 15k high-quality human-generated prompt/response pairs specifically designed for instruction tuning LLMs. The dataset was authored by more than 5,000 Databricks[6] employees during March and April of 2023, and instructions are not copied from the web or other LLMs.
- BactrianX[7] (Li et al. 2023) is a collection of 3.4M instruction-response pairs in 52 languages, that are obtained by translating 67K English instructions (alpaca + dolly) into 51 languages using Google Translate API. The translated instructions are then fed to ChatGPT (get-3.5-turbo) to obtain its natural responses, resulting in 3.4M instruction-response pairs in 52 languages, including Italian.
- EVALITA (Lai et al. 2023) is an evaluation campaign of NLP and speech tools for the Italian language. EVALITA has been held every two years since 2007, and, over the years, many tasks have been released, greatly contributing to the Italian research on NLP. We have focused on the last edition that took place in 2023, selecting some of the tasks that have been proposed.

The adaptation and fine-tuning process is sketched in Figure 1. In detail, starting from the BLOOM-1b7 model, we obtain four fine-tuned models: one for each instruction dataset (Camoscio, Dolly, and BactrianX) plus the EVALITA model. Then, the BLOOM-1b7 model is adapted to Italian, leveraging data from the Italian corpus and obtaining the Italian-adapted model called BLOOM-IT-1b7. This adapted model is fine-tuned on all the instruction and EVALITA datasets, resulting in another four other fine-tuned models, but this time, they are constructed by fine-tuning the Italian-adapted model. Therefore, at the end of the whole pipeline, there is a total of 8 models. For the sake of simplicity, in the figure and throughout the paper, we remove the suffix "-1b7" and simply use "BLOOM" to refer to the BLOOM-1b7 model.

2 https://huggingface.co/bigscience/bloom-1b7
3 We use a cleaned version of the Camoscio dataset: https://huggingface.co/datasets/teelinsan/camoscio_cleaned
4 https://huggingface.co/datasets/basilepp19/dolly-15k-it
5 https://huggingface.co/datasets/databricks/databricks-dolly-15k
6 https://www.databricks.com/
7 https://huggingface.co/datasets/MBZUAI/Bactrian-X

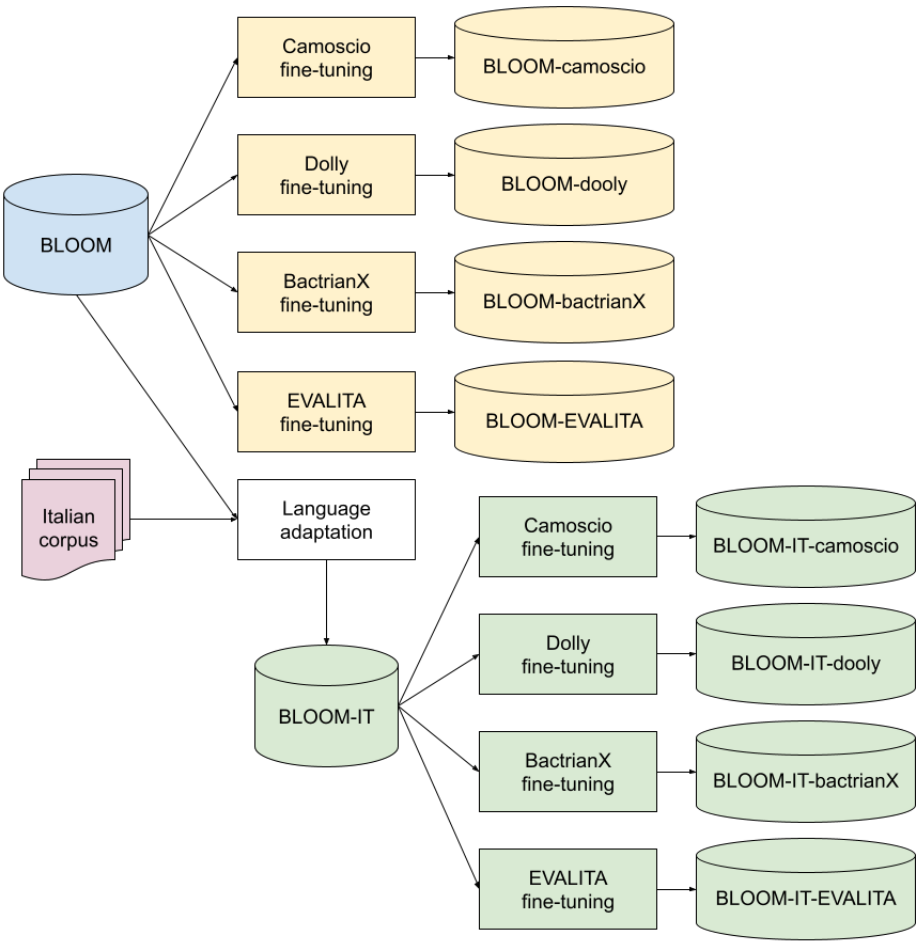

Figure 1
The adaptation pipeline

We decided to use the 1b7 version of the BLOOM model to reduce computational costs. It has not been trained on any dialogue instruction like its counterpart, BLOOMZ, and it does not contain the training data documents written in Italian. We follow the hypothesis that instruction-based fine-tuning should be performed after a phase of language adaptation, with instructions provided in the specific language of interest.

As a language adaptation strategy, we use continuing the pre-training on new language-specific data by relying on PEFT and LoRA to reduce the number of trainable passages. For the adaptation step, we use a corpus of Italian texts extracted from the March 2024 dumps of Italian Wikipedia, Wikinews, and Wikibooks to avoid any issues with intellectual property and copyrighted material. The final corpus consists of 2,899,019 documents. During the adaptation step, all documents are split to fulfil the length requirement of 512 tokens. This size was chosen to guarantee an adequate batch size during the adaptation step without exceeding the GPU's memory limit.

The instruction datasets are mostly about Open/Closed Q&A, ExtractSummarize information, Brainstorming, Classification, and Creative writing.

Finally, we opt to fine-tune the models over data from EVALITA 2023 tasks. We have gathered the training and test data from the tasks' websites when available or directly contacted the task organizers in case they have chosen not to disclose such information publicly.

3.1 Implementation details

We adopt the following libraries for the training: Transformers, PEFT, and TRL. As a PEFT strategy, we adopt LoRA with $r = 16$ and $lora\_alpha = 32$. For other LoRA parameters, we use the default values.

For the adaptation step, we train the model for 3 epochs with a batch size of 8 and one gradient accumulation step using an input length of 512. For the fine-tuning of instruction models, we train each model for 10 epochs with a batch size of 4 and 4 gradient accumulation steps using an input length of 1024.

For all the models we use the `adamw_bnb_8bit` optimizer with the following parameters: $learning\_rate = 2e - 4$, $max\_grad\_norm = 0.3$ and $warmup\_ratio = 0.03$.

All the models are trained on a single NVIDIA-RTX A6000 GPU with 48GB of memory. The adaptation step requires about four days, while Camoscio and Dolly require less than 15 hours. BactrianX and EVALITA require about one day. All the code is available on GitHub[8].

3.2 Data Release

Following the open-science principles, we release nine models on HuggingFace[9]. The available models are:

- **BLOOM-camoscio, BLOOM-dolly, BLOOM-bactrianx**: the BLOOM models fine-tuned on instruction datasets;
- **BLOOM-IT**: the Italian adapted version of BLOOM using the Italian corpus;
- **BLOOM-IT-camoscio, BLOOM-IT-dolly, BLOOM-IT-bactrianx**: the BLOOM-IT models fine-tuned on instruction datasets;
- **BLOOM-EVALITA, BLOOM-IT-EVALITA**: the two models obtained by fine-tuning BLOOM and BLOOM-IT on the instruction dataset built on the EVALITA task.

4. Validation and Discussion of Results

We perform two kinds of evaluation. The first one aims to evaluate each model on standard benchmarks for assessing LLMs' ability to understand human language. The second one evaluates the performance of EVALITA models on each task.

8 https://github.com/swapUniba/bloom-it
9 https://huggingface.co/collections/swap-uniba/bloom-it-668be2b26437930851afae40

4.1 Language Model Evaluation Harness

Language Model Evaluation Harness[10] provides a unified framework for testing generative language models on a variety of evaluation tasks. The tool includes over 60 standard academic benchmarks for LLMs, with hundreds of subtasks and variants. Evaluation with publicly available tools and prompts ensures reproducibility and comparability between models. It also supports custom prompts and evaluation metrics. Moreover, the Language Model Evaluation Harness is the back-end for HuggingFace's popular Open LLM Leaderboard and includes several benchmarks automatically translated into Italian. Currently, the tool is used to maintain an Open Italian LLM Leaderboard available on HuggingFace[11].

Table 1
Results on the BLOOM models obtained by the Language Model Evaluation Harness. This table considers only benchmarks adopted by the Open Italian LLM Leaderboard. **Avg** is the average of the three tasks.

model	**hellaswag**	**arc**	**m_mmlu**	**avg**
bloom	*.4796*	*.2671*	*.2702*	*.3390*
model	**hellaswag_it**	**arc_it**	**m_mmlu_it**	**avg**
bloom	.3351	.2344	**.2704**	.2800
bloom-camoscio	.3336	.2489	.2540	.2788
bloom-dolly	.3343	.2464	.2674	.2827
bloom-bactrianx	**.3457**	.2275	.2618	.2783
bloom-evalita	.3289	.2258	.2425	.2657
bloom-it	.3385	.2429	.2529	.2781
bloom-it-camoscio	.3436	.2481	.2567	.2828
bloom–it-dolly	.3400	**.2549**	.2670	**.2873**
bloom-it-bactrianx	.3442	.2498	.2664	.2868
bloom-it-evalita	.3334	.2387	.2582	.2768

Results in Table 1 show the performance in terms of accuracy of all the models on the three tasks selected by the Open Italian LLM Leaderboard. The **avg** column is computed by averaging the score of all the tasks. The first row reports the BLOOM model's performance on the English datasets, while all other results are computed on the Italian-translated version of each benchmark. The MMLU evaluation uses five examples in the few-shot context, and all the evaluations are executed with a batch size equal to 2.

The involved benchmarks are:

- **HellaSWAG** is a dataset for studying grounded commonsense inference. It consists of 70k multiple-choice questions about grounded situations. Each question comes from one of two domains (activitynet or wikihow) with four answer choices about what might happen next in the scene. The correct answer is the (real) sentence for the next event; the three incorrect answers are adversarially generated and human-verified.

10 https://github.com/EleutherAI/lm-evaluation-harness
11 https://huggingface.co/spaces/FinancialSupport/open_ita_llm_leaderboard

- The AI2's Reasoning Challenge (**ARC**) dataset is a multiple-choice question-answering dataset containing questions from science exams from grade 3 to grade 9.
- **MMLU** (Massive Multitask Language Understanding) is a benchmark designed to measure knowledge acquired during pretraining by evaluating models exclusively in zero-shot and few-shot settings. This makes the benchmark more challenging and more similar to how humans are evaluated. The benchmark covers 57 subjects across STEM, the humanities, the social sciences, and more.

Analyzing the results, we observe that the model's performance on the Italian datasets is similar to that on the English datasets except for the HellaSWAG benchmark. The language adaptation step seems to consistently improve performance on the ARC dataset, while performances on the other benchmarks are similar, with some decrease in the MMLU benchmark. However, the BLOOM-IT model fine-tuned on the dolly dataset obtains the best overall result. The BLOOM-IT model is able to overcome the BLOOM one on HellaSWAG and ARC, but a drop in the MMLU performance results in a worse overall score. This phenomenon can be explained by the fact that by adopting LoRA, we trained on a limited number of parameters on a small model. In future work, we plan to investigate the performance of full parameter optimization. This issue also affects the adapted model (BLOOM-it).

We perform a further evaluation by exploiting other Italian benchmarks not included in the Open Italian LLM Leaderboard. In detail, we take into account other three datasets:

- **lambada**: an open-ended cloze task which consists of about 10,000 passages from BooksCorpus where a missing target word is predicted in the last sentence of each passage.
- **xcopa**: the Choice Of Plausible Alternatives (COPA) evaluation provides a tool for assessing LLMs performance in open-domain commonsense causal reasoning.
- **belebele**: a multiple-choice machine reading comprehension (MRC) dataset spanning 122 language variants.

Over this benchmark, lambada is particularly interesting because it is the only one not based on multiple-choice QA and can be used to test the quality of the generated text; in fact, it is the only one that also adopts perplexity (*lambada (P)*) as a metric.

We observe that the adapted models always perform better. Surprisingly, the adapted model fine-tuned on the BactrianX dataset is able to overcome the original BLOOM model on the English version of the belebele dataset. Another very relevant result is the perplexity obtained by the BLOOM-it model with respect to the one obtained by the BLOOM model when it is tested on the Italian version of lambada datasets. The drop in perplexity proves that the Italian adaptation is crucial in text generation quality benchmarks.

It is important to underline that in both evaluations, the models based on EVALITA generally achieve the worst performance since they are instructed to generate answers according to the EVALITA tasks guidelines, which are different from the benchmarks

used in this evaluation. A specific evaluation of EVALITA tasks is reported in Section 4.2.

Table 2
Results on the BLOOM models obtained by the Language Model Evaluation Harness. This table considers other Italian benchmarks not included in the Open Italian LLM Leaderboard.

model	**lambada (P)**	**lambada (A)**	**xcopa**	**belebele**
bloom	12.59	.4630	.5505	.2378
model	**lambada_it (P)**	**lambada_it (A)**	**xcopa_it**	**belebele_it**
bloom	691.36	.2274	.5260	.2278
bloom-camoscio	967.08	.2298	.5180	.2367
bloom-dolly	428.73	.2661	.5240	.2378
bloom-bactrianx	400.13	.2686	.5160	.2211
bloom-evalita	867.61	.2076	.5460	.2289
bloom-it	**351.00**	**.2734**	.5540	.2189
bloom-it-camoscio	682.74	.2375	**.5580**	.2356
bloom-it-dolly	403.04	.2645	.5340	.2378
bloom-it-bactrianx	452.10	.2408	.5280	**.2578**
bloom-it-evalita	791.44	.2206	.5320	.2267

4.2 EVALITA evaluation

For the evaluation of the zero-shot abilities of the models fine-tuned on EVALITA, we select the following tasks of the last 2023 edition:

- DisCoTEX – Assessing DIScourse COherence in Italian TEXts
- EMit – Categorical Emotion Detection in Italian Social Media
- HaSpeeDe – Political and Religious Hate Speech Detection
- HODI – Homotransphobia Detection in Italian
- NERMuD - Named-Entities Recognition on Multi-Domain Documents
- PoliticIT – Political Ideology Detection in Italian Texts
- WiC-ITA – Word-in-Context task for Italian

Other tasks were left out of the evaluation due to the lack of training/test data or the nature of the tasks themselves, for example tasks for which the prompt always exceeds the maximum input length of the models. As stated in Section 3, we fine-tuned the BLOOM and BLOOM-IT on the aforementioned tasks, thus obtaining the BLOOM-EVALITA and BLOOM-IT-EVALITA models. The fine-tuning was performed using the following prompt:

```
Di seguito è riportata un'istruzione che descrive un'attività,
abbinata ad un input che fornisce ulteriore informazione.
Scrivi una risposta che soddisfi adeguatamente la richiesta.
### Istruzione:{instruction}
### Input:{context}
```

```
### Risposta:{response}
```

Where `{instruction}` is a sentence describing the task the model must complete, `{context}` contains the EVALITA task entry while `{response}` is the output of the LLM. Clearly, the instructions are heavily dependent on the task and have to be designed so that they can be solved by a generative model. A complete list of the instructions for each task is given in Table 3. As the tasks have been reformulated, the model's output does not directly match those of the original tasks, so they must be converted back to be evaluated, as we chose to use the original evaluation scripts for each task where available (we refer to the respective task description papers for details on how each metric is computed). Some cases are quite straightforward, like for the binary tasks where the output (*si/no*) can be easily mapped to a binary value, while some are quite more complex, like for the NERMuD task where the LLM is asked to list all the Named Entities that appear in the input text preceded by their type (i.e. `LOC`, `PER`, and `ORG`) while the output required for the evaluation is a tab-separated file in the IOB format.

Table 4 shows the results obtained by the two models on each task, compared with the respective baselines and the results obtained by ExtremITA (Hromei et al. 2023). In particular, we have considered the LLaMA version of ExtremITA (i.e., a 7b parameters model built upon LLaMA), which participated in the last edition of EVALITA and obtained outstanding results in many tasks. The results shown in the table highlight how much the number of parameters can affect the model's performance. There are remarkable differences in the scores obtained by BLOOM (1.7b parameters) and ExtremITA (7b parameters). Comparing the results obtained by the two versions of BLOOM, we can see that language adaptation usually slightly improves the model's results. Probably, this is due to the PEFT technique in which only a portion of the model's parameters are trained thus compromising the overall performance. Furthermore, there are instances where the model's performance falls short of the baseline. This underscores the need for nuanced considerations when employing smaller language models in certain contexts.

5. Conclusions

This paper explores a language adaptation strategy for the BLOOM model to address the challenge of handling languages not covered during the training. Despite the remarkable capabilities of the BLOOM model in understanding natural language for widely spoken languages, it showed limitations when applied to languages which are not included in the original training set, such as Italian. To overcome this limitation, we conducted experiments using a language adaptation step based on continuing the pre-training on Italian documents followed by instruction-based fine-tuning on Italian data.

We use the Language Model Evaluation Harness tool to test the obtained models on several benchmarks for assessing LLMs' ability. Results show that language adaptation generally improves the BLOOM model's ability to generate text in Italian, especially on the lambada dataset in which the LLM should generate text, and the original BLOOM model cannot generate Italian text.

We also evaluated several EVALITA 2023 datasets, which highlighted the importance of balancing the number of parameters and the fine-tuning techniques of choice. This further suggests the need for investigating the suitability of smaller language models in specific contexts. Another research direction could be testing the performance

Table 3
Instruction for the EVALITA task considering in the fine tuning step. Note that text in curled brackets represents a placeholder for information that is added to the instruction based on the task entry.

Task	Subtask	Instruction
DisCoTEX	A	Classifica la frase in input come 'Coerente' se si integra logicamente e contribuisce a formare un testo coerente con il paragrafo di contesto. Se la frase target risulta incoerente con il paragrafo, classificala come 'Incoerente'.
	B	Predici il punteggio medio di coerenza assegnato dai valutatori umani per il testo in input. Utilizza una scala ordinale a 5 punti (da 1 a 5) per riflettere la percezione graduale della coerenza.
EMit	A	Categorizza le emozioni espresse nel testo fornito in input o determina l'assenza di emozioni. Puoi classificare il testo come neutrale o identificare una o più delle seguenti emozioni: rabbia, anticipazione, disgusto, paura, gioia, tristezza, sorpresa, fiducia, amore.
	B	Classifica il testo fornito identificando se il target del messaggio si riferisce all'argomento ('topic'), alla direzione ('direction'), a entrambi ('both') o a nessuno ('none') dei due. Considera la natura del testo e il contesto degli spettacoli televisivi e musicali per determinare il target appropriato.
HODI	A	Stabilisci se il testo in input ha contenuti omotransfobici o meno. Rispondi con si o no.
	B	Estrai dal testo in input le parole che denotano concetti omotransfobici. Separa le parole estratte con [SEP]. Se ci sono parole estratte, restituisci 'Non omotransfobico'.
NERMuD	DAC	Elenca le menzioni di entità presenti nel testo in input, indicandone il tipo: [PER] (persona), [LOC] (luogo), [ORG] (organizzazione). Se non ci sono entità, resituisci: 'Nessuna menzione'
PoliticIT		"Indica se l'autore del testo in input è un 'uomo' o una 'donna', seguito dalla sua appartenenza politica scegliendo tra 'destra', 'sinistra', 'centrodestra', 'centrosinistra'.
wicITA	binary	Stabilisci nelle due frasi in input la parola '{target}' è usata con lo stesso significato. Rispondi con si o no.
	ranking	Predici il punteggio medio assegnato dai valutatori umani per indicare quanto simile è il significato della parola '{target}' nelle due frasi in input. Utilizza una scala ordinale a 4 punti (da 1 a 4) per riflettere la percezione graduale della similarità del significato.
HaSpeeDe	textual	Stabilisci se il tweet in input contiene discorsi che incitano all'odio. Rispondi con si o no.
	contextual	Stabilisci se il tweet in input contiene discorsi che incitano all'odio considerando anche il contesto relativo alle statistiche dell'account. Rispondi con si o no. Contesto: Data: {data} Numero di retweet: {number of retweets} Numero di mi piace: {number of likes} Data creazione account: {account creation date} Numero di post: {number of posts} Follower: {number of followers} Amici: {number of friends}

of adapted models with a larger number of parameters such as LLaMAntino (Basile et al. 2023b; Polignano, Basile, and Semeraro 2024).

The proposed methodology can be adapted for other languages following the same pipeline adopted in this work. The adapted models can be easily fine-tuned on several tasks, providing proper instructions. Our future research will extend to testing this

Table 4
Results on the EVALITA 2023 tasks.

Task	Subtask	Metric	bloom-evalita	bloom-it-evalita	extremITA	Baseline
DisCoTEX	A	Acc	0.6919	0.6938	0.8150	0.525
	B	HM	0.3159	0.3904	0.6500	0.11
EMit	A	F1	0.4005	0.3998	0.6028	0.4074
	B	F1	0.6194	0.6357	0.6459	0.6184
HODI	A	F1	0.6614	0.6485	0.7942	0.51
	B	agreement	0.5660	0.5643	0.7228	0.6691
NERMuD		macroF1	0.78	0.78	0.8900	0.83
PoliticIT		F1	0.5531	0.5563	0.7719	0.569
wicITA	1	F1	0.3605	0.4087	0.5100	0.594
	2	spearman	0.0608	0.1046	0.5100	0.569
HaSpeeDe	A-textual	F1	0.7629	0.7700	0.9034	0.8457
	A-contextual	F1	0.5085	0.5156	0.9034	0.8457

approach with other languages and open models, contributing to the broader landscape of language model adaptability.

Acknowledgments
We acknowledge the support of the PNRR project FAIR - Future AI Research (PE00000013), Spoke 6 - Symbiotic AI (CUP H97G22000210007) under the NRRP MUR program funded by the NextGenerationEU.

References

Ansell, Alan, Edoardo Maria Ponti, Anna Korhonen, and Ivan Vulic. 2022. Composable Sparse Fine-Tuning for Cross-Lingual Transfer. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 1778–1796. Association for Computational Linguistics.

Basile, Pierpaolo, Pierluigi Cassotti, Marco Polignano, Lucia Siciliani, and Giovanni Semeraro. 2023a. On the Impact of Language Adaptation for Large Language Models: A Case Study for the Italian Language Using Only Open Resources. In Federico Boschetti, Gianluca E. Lebani, Bernardo Magnini, and Nicole Novielli, editors, *Proceedings of the 9th Italian Conference on Computational Linguistics, Venice, Italy, November 30 - December 2, 2023*, volume 3596 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Basile, Pierpaolo, Elio Musacchio, Marco Polignano, Lucia Siciliani, Giuseppe Fiameni, and Giovanni Semeraro. 2023b. LLaMAntino: LLaMA 2 models for effective text generation in Italian language. *arXiv preprint arXiv:2312.09993*.

Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Chau, Ethan C., Lucy H. Lin, and Noah A. Smith. 2020. Parsing with Multilingual BERT, a Small Treebank, and a Small Corpus. In Trevor Cohn, Yulan He, and Yang Liu, editors, *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 1324–1334. Association for Computational Linguistics.

Chiang, Wei-Lin, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*.

Conover, Mike, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. 2023. Free Dolly: Introducing the World's First Truly Open Instruction-tuned LLM. *Company Blog of Databricks*.

Guo, Yunhui. 2018. A Survey on Methods and Theories of Quantized Neural Networks. *arXiv preprint arXiv:1808.04752*.

Hromei, Claudiu D., Danilo Croce, Valerio Basile, and Roberto Basili. 2023. ExtremITA at EVALITA 2023: Multi-Task Sustainable Scaling to Large Language Models at its Extreme. In Mirko Lai, Stefano Menini, Marco Polignano, Valentina Russo, Rachele Sprugnoli, and Giulia Venturi, editors, *Proceedings of the Eighth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2023), Parma, Italy, September 7th-8th, 2023*, volume 3473 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Hu, Edward J., Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Hu, Zhiqiang, Lei Wang, Yihuai Lan, Wanyu Xu, Ee-Peng Lim, Lidong Bing, Xing Xu, Soujanya Poria, and Roy Ka-Wei Lee. 2023. LLM-Adapters: An Adapter Family for Parameter-Efficient Fine-Tuning of Large Language Models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 5254–5276. Association for Computational Linguistics.

Jiao, Xiaoqi, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. Tinybert: Distilling BERT for natural language understanding. In Trevor Cohn, Yulan He, and Yang Liu, editors, *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 4163–4174. Association for Computational Linguistics.

Lai, Mirko, Stefano Menini, Marco Polignano, Valentina Russo, Rachele Sprugnoli, and Giulia Venturi. 2023. EVALITA 2023: Overview of the 8th evaluation campaign of natural language processing and speech tools for italian. In Mirko Lai, Stefano Menini, Marco Polignano, Valentina Russo, Rachele Sprugnoli, and Giulia Venturi, editors, *Proceedings of the Eighth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2023), Parma, Italy, September 7th-8th, 2023*, volume 3473 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Li, Haonan, Fajri Koto, Minghao Wu, Alham Fikri Aji, and Timothy Baldwin. 2023. Bactrian-X : A Multilingual Replicable Instruction-Following Model with Low-Rank Adaptation. *arXiv preprint arXiv:2305.15011*.

Nag, Arijit, Bidisha Samanta, Animesh Mukherjee, Niloy Ganguly, and Soumen Chakrabarti. 2023. Entropy-guided Vocabulary Augmentation of Multilingual Language Models for Low-resource Tasks. In Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 8619–8629. Association for Computational Linguistics.

Naveed, Humza, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Nick Barnes, and Ajmal Mian. 2023. A Comprehensive Overview of Large Language Models. *arXiv preprint arXiv:2307.06435*.

Nowakowski, Karol, Michal Ptaszynski, Kyoko Murasaki, and Jagna Nieuważny. 2023. Adapting Multilingual Speech Representation Model for a New, Underresourced Language through Multilingual Fine-tuning and Continued Pretraining. *Information Processing & Management*, 60(2):103148.

Polignano, Marco, Pierpaolo Basile, and Giovanni Semeraro. 2024. Advanced Natural-based interaction for the ITAlian language: LLaMAntino-3-ANITA. *arXiv preprint arXiv:2405.07101*.

Santilli, Andrea and Emanuele Rodolà. 2023. Camoscio: An Italian Instruction-tuned LLaMA. In Federico Boschetti, Gianluca E. Lebani, Bernardo Magnini, and Nicole Novielli, editors, *Proceedings of the 9th Italian Conference on Computational Linguistics, Venice, Italy, November 30 - December 2, 2023*, volume 3596 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Scao, Teven Le, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2022. BLOOM: A 176B-Parameter Open-Access Multilingual Language Model. *arXiv preprint arXiv:2211.05100*.

Touvron, Hugo, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. LLaMA: Open and Efficient Foundation Language Models. *arXiv preprint arXiv:2302.13971*.

Wang, Hai, Dian Yu, Kai Sun, Jianshu Chen, and Dong Yu. 2019. Improving Pre-Trained Multilingual Model with Vocabulary Expansion. In Mohit Bansal and Aline Villavicencio, editors, *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 316–327, Hong Kong, China, November. Association for Computational Linguistics.

Wang, Ruize, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Jianshu Ji, Guihong Cao, Daxin Jiang, and Ming Zhou. 2021. K-Adapter: Infusing Knowledge into Pre-Trained Models with Adapters. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 1405–1418. Association for Computational Linguistics.

Wang, Yizhong, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-Instruct: Aligning Language Models with Self-Generated Instructions. In Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 13484–13508. Association for Computational Linguistics.

Yong, Zheng Xin, Hailey Schoelkopf, Niklas Muennighoff, Alham Fikri Aji, David Ifeoluwa Adelani, Khalid Almubarak, M. Saiful Bari, Lintang Sutawika, Jungo Kasai, Ahmed Baruwa, Genta Indra Winata, Stella Biderman, Edward Raff, Dragomir Radev, and Vassilina Nikoulina. 2023. BLOOM+1: adding language support to BLOOM for zero-shot prompting. In Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 11682–11703. Association for Computational Linguistics.