Extending Italian Large Language Models for vision-language tasks

Elio Musacchio[1,2,*], Lucia Siciliani[1], Pierpaolo Basile[1], Asia Beatrice Uboldi[3], Giovanni Germani[3] and Giovanni Semeraro[1]

[1]Department of Computer Science, University of Bari Aldo Moro, Italy
[2]National PhD in Artificial Intelligence, University of Pisa, Italy
[3]Fastweb SpA, Milan, Italy

Abstract

With the growing evolution of Large Language Models, there has also been a rising interest in extending these models to incorporate non-textual signals. Specifically, Large Vision-Language Models have been developed, which extend Large Language Models to understand and process visual signals. This allows them to solve complex vision-language tasks, further extending their inherent abilities in text-only task resolution. However, for the Italian language, most works still focus on text-only solutions without extending them to multimodality. In this work, we extend Large Language Models for the Italian language to multimodality and benchmark the performance of these models when trained using the same experimental setting.

Keywords

Large Language Models, Large Vision-Language Models, Multimodality

1. Introduction

In the last years, interest in Large Language Models (LLMs) has been growing steadily. The ability of these models to solve complex tasks, even when they have not been trained with that specific objective, makes them extremely useful for any natural language processing task. However, as it often occurs in the Natural Language Processing research field, the abundance of English data meant that the first openly released LLMs only supported the English language (e.g. LLaMA 2 [1]), limiting the applicability of these models to other languages. To cover this gap, several LLMs were trained to directly support the Italian language, using either a monolingual or multilingual strategy. Whichever the selected strategy, these models were obtained using one of the following methodologies: fine-tuning pre-existing models or training from scratch on datasets consisting mainly of Italian data. This trend allowed to extend LLMs not only to multiple underrepresented languages but also to new modalities. An example is represented by Large Vision-Language Models (LVLMs) that are LLMs extended with a technique enabling them to process visual inputs together with textual ones. Also in this case, there are training procedures that allow leveraging existing LLMs instead of training from scratch for vision-language inputs. This makes the process both more efficient, since the pre-training phase is skipped, and more effective, as the textual knowledge of the model is leveraged to learn how to perform vision-language tasks. Despite this, many open LLMs supporting the Italian language have not been extended to support multimodality. This is due to the limited availability of training data for vision-language tasks in Italian, whereas English training data often comprises multiple diverse and rich tasks. Furthermore, with the proliferation of Italian LLMs, like Minerva [2] and Velvet[1], it becomes increasingly important to test their capabilities in the multimodal domain. This raises the question of whether it is possible to extend current LLMs trained for the Italian language for multimodality. Do these models perform well when extended to support it? In this work, we propose a study on the multimodal performance of Italian LLMs extended to LVLMs using a state-of-the-art approach.

Specifically, this work extends current literature as follows:

- We train several LLMs supporting the Italian language to extend them to LVLMs;
- We benchmark these models using datasets that are natively in Italian;
- We study the effect of different prompt formatting at inference time and showcase the length bias in the response of LVLMs.

CLiC-it 2025: Eleventh Italian Conference on Computational Linguistics, September 24 — 26, 2025, Cagliari, Italy
*Corresponding author.
elio.musacchio@uniba.it (E. Musacchio); lucia.siciliani@uniba.it (L. Siciliani); pierpaolo.basile@uniba.it (P. Basile); asiabeatrice.uboldi@consulenti.fastweb.it (A. B. Uboldi); giovanni.germani@fastweb.it (G. Germani); giovanni.semeraro@uniba.it (G. Semeraro)
0009-0006-9670-9998 (E. Musacchio); 0000-0002-1438-280X (L. Siciliani); 0000-0002-0545-1105 (P. Basile); 0000-0001-6883-1853 (G. Semeraro)

[1]https://huggingface.co/Almawave/Velvet-14B

Finally, we want to underline that we are forced to use machine translation for training data due to the scarcity of large-scale multimodal data for non-English languages. However, we focus our evaluation on natively Italian multimodal datasets. Therefore, if a large-scale multimodal dataset natively in Italian were to be released, we can expect further improvements in performance since fewer machine translation errors would be present.

Furthermore, we release code and resources related to this study[2].

2. Related Works

For LVLMs, several methodologies have been designed to adapt LLMs. One of the most prominent approaches is the one introduced in LLaVA [3], where visual embeddings extracted from a Vision Transformer [4] are projected into the latent space of an LLM. This strategy has been further refined in LLaVA 1.5 [5], where the projection matrix is replaced with a Multi-Layer Perceptron, and LLaVA-OneVision [6], a LLaVA-based model enhanced to also perform multi-image and video tasks. Other approaches include the one used in BLIP-2 [7], leveraging a QFormer module to extract the most relevant features of images, and Flamingo [8], where cross-attention layers are added to the LLM and relevant visual tokens are extracted using a Perceiver Resampler module. Additionally, there is also LLaVA-NeXT [5] (also known as LLaVA 1.5 HD), which introduces a technique to process high-resolution images. The idea is to resize the image to a higher resolution than the one supported by the underlying vision encoder and split it into multiple images. Embeddings are then extracted for each image, as well as a resized version of the image to the supported resolution of the vision encoder to incorporate global details, and flattened into a single vector.

For Italian LLMs, several models have been released which incorporate a great quantity of natively Italian training data. Minerva [2] is the first family of models trained from scratch on an open data mixture consisting of only English and Italian data. It has several checkpoints with different parameter counts, that are 1B, 3B and 7B. The 7B model was trained on a total of 2.48 trillion tokens of Italian, English and code. EuroLLM [9] is a family of LLMs developed in Europe to support all the 24 official European Union languages. Its two available checkpoints have 1.7B and 9B parameters. The models are pre-trained on a total of 4 trillion tokens, where 50% of the data is in English, 5% is code, and the remaining 45% are other languages (including Italian). Velvet is a family of LLMs trained on a balanced mixture of six languages, with particular emphasis on Italian (which makes 23% of the data). The two available checkpoints for Velvet have 2B and 14B parameters.

FastwebMIIA[3] (Italian Artificial Intelligence Model) is a 7-billion-parameter autoregressive model developed by Fastweb. Based on a decoder-only architecture with rotary positional embeddings, it has been trained on about 3 trillion tokens, with a strong focus on Italian. It uses a custom tokenizer optimised for Italian, English and programming languages, with a vocabulary of 50,000 tokens. It supports a context window of 16k tokens and has been trained in a distributed pipeline on NVIDIA H100 GPUs via MLDE and LLMFoundry.

Furthermore, at the time of writing, LLaVA-NDiNO [10] is the only family of multimodal models extensively trained for the Italian language only, further showcasing the need for a more in-depth investigation of the current landscape of Italian LLMs and their extension to LVLMs.

For LLM evaluation in Italian, many efforts have been carried out to extensively evaluate Italian LLMs. For example, Bacciu et al. [11] introduced an open LLM leaderboard for the Italian language, Moroni et al. [12] released ITA-Bench, a comprehensive evaluation suite for Italian LLMs consisting of both machine-translated and natively Italian benchmarks, Attanasio et al. [13] released CALAMITA, a dynamic and growing benchmark for the Italian language.

Finally, we also highlight that there are novel works that showcase how non-trivial it is to evaluate LLMs. For example, Wang et al. [14] found mismatches between the generated output and output obtained using log-likelihood for next token prediction. Additionally, several works started to use a LLM-as-a-judge approach where the LLM is used as a model for evaluation [15].

3. Methodology

As mentioned in the introduction, our aim is to extend existing Italian LLMs with multimodal capabilities. We

Table 1
Training hyperparameters used for the two steps of the LLaVA NeXT methodology

Parameter	Stage	
	Pre-Train	Fine-Tune
Batch Size	256	128
Learning Rate	1e-3	1e-5
Weight Decay	0.	0.
Warmup Ratio	0.03	0.03
Epochs	1	1

[2]https://github.com/swapUniba/Extending-LLMs-VL-ITA

[3]https://huggingface.co/Fastweb/FastwebMIIA-7B

Listing 1: Mistral chat template used for base models. {user} and {assistant} are placeholders for the user and assistant messages respectively.

```
<s>[INST] {user} [/INST] {assistant}</s>
```

chose Minerva, EuroLLM, Velvet and FastwebMIIA, since they are among the most recently released LLMs supporting the Italian language and clearly define the amount of Italian data used in training. For each model, we evaluate both its base and instruct variants at their largest available parameter scale. The only exception is represented by Velvet, for which only the instruct version is available.

For the vision backbone, we use the vision transformer of the CLIP [16] model, specifically, we focus on the large checkpoint with patch size 14 and image size 336.[4] We use this model since it is often used in the state-of-the-art research as the visual backbone for LVLMs [3].

To train the models, we use the methodology of LLaVA NeXT, because of both its performance and its open codebase, which allows for easier reproducibility of this study. This methodology is made of two steps: *pre-training* to warm up the multi-layer perceptron projector and *visual instruction tuning* to teach the model how to solve vision-language tasks. For both steps, training is performed using the next token prediction objective, implemented as cross-entropy loss. We report hyperparameters used for both steps in Table 1. For base models, we apply the Mistral chat template reported in Listing 1, since they do not have a chat template associated with them, while for instruct models we apply their own chat template.

3.1. Training Mixture

For both training steps, we use a state-of-the-art machine translation model to translate popular vision-language English-only datasets to Italian. This is necessary since, at the time of writing, there is no large-scale vision-language dataset for instruction tuning in Italian. Therefore, we use MADLAD 400 3B [17], since it is one of the latest and best-performing machine translation models. For *pre-training*, we use the same dataset as LLaVA translated to the Italian language. During *pre-training*, the whole model is kept frozen, except for the multi-layer perceptron. Thanks to this approach, the multi-layer perceptron weights are initialized so that the vision embeddings are correctly projected into the LLM's space. For *visual instruction tuning*, we consider a combination of two datasets: *MultiInstruct* [18] and the *conversational* split of the LLaVA-Instruct [3] dataset. The former is a collection of diverse vision-language tasks (e.g. Visual Question Answering, Visual Grounding, ...), which allows the model to learn to correctly solve this type of task, while the latter is a multi-turn dataset generated by prompting GPT-4. Thanks to this training mixture, the model learns to both solve tasks and provide meaningful and complete responses to user prompts. For MultiInstruct, we perform some additional processing operations. Instructions are manually translated, therefore only the data instances (e.g. questions and answers in a visual question-answer task) are machine-translated. For tasks that use bounding boxes, we normalize the bounding box values to the [0, 1] range so that the values are consistent with the reference images and independent of their resolution. For tasks that provide options to choose from within the instruction, we format them as an ordered list using either numbers, uppercase or lowercase letters, or plain text. In such cases, we also replace the target text to be predicted with the corresponding identifier (e.g. if the option is a number, the target text is also converted to a number). Finally, we append a string to guide model responses, depending on the type of output that is expected: "Rispondi solamente con il numero dell'opzione corretta dalle scelte date." ("Answer with the option's number from the given choices directly." in English) when the options are identified by numbers, "Rispondi solamente con la lettera dell'opzione corretta dalle scelte date." ("Answer with the option's letter from the given choices directly." in English) when the options are identified by letters, "Rispondi usando una zona di delimitazione." ("Answer using a bounding box." in English) when the target text is a bounding box and, finally, "Rispondi usando una singola parola o frase." ("Answer the question using a single word or phrase." in English) for all other cases. In total, the training mixture combining these two datasets consists of 172,335 instances.

3.2. Hardware and Software Configuration

Our experimental setup was provided by Fastweb SpA via a high-performance computing cluster [5] composed of 31 NVIDIA DGX H100 systems, organized according to the NVIDIA DGX SuperPOD reference architecture. The cluster is deployed within a datacenter located in Lombardia, Italy, and offers a total of 248 NVIDIA H100 Tensor Core GPUs interconnected through high-bandwidth NVLink and InfiniBand, enabling low-latency communication and efficient scaling across nodes.

The training and evaluation of the models was conducted in a distributed manner through the Machine Learning Distributed Engine *MLDE*[6] platform, which enabled efficient parallelisation of workloads on DGX H100

[4] https://huggingface.co/openai/clip-vit-large-patch14-336

[5] Fastweb Announcement

[6] https://www.hpe.com/us/en/software/marketplace/hpe-ml-development-environment.html

Table 2
We report results for the three benchmarks considered in this study. All benchmarks are evaluated using *exact match* as metric. The best result for each dataset and each formatting is in bold.

Dataset	Type	Model	Formatting				AVG
			Pre	Post	Pre-Swap	Post-Swap	
GQA-IT	base	Minerva-7B	.2867	.3523	.2523	.2023	.2734
		EuroLLM-9B	.2893	.3917	.4157	.0973	**.2985**
		FastwebMIIA-7B	.1683	**.4147**	.4043	.0297	.2543
	instruct	Minerva-7B	.2653	.3520	.3120	.0533	.2457
		EuroLLM-9B	.0370	.4140	**.4187**	.0677	.2344
		Velvet-14B	**.3007**	.2863	.3107	**.2843**	.2955
		FastwebMIIA-7B	.0933	.3233	.3227	.0790	.2046
		LLaVA-NeXT 8B	.0009	.3106	.3454	.0849	.1855
MTVQA-IT	base	Minerva-7B	.0486	.0577	.0509	.0373	.0486
		EuroLLM-9B	**.1018**	.1097	.1143	.0792	.1013
		FastwebMIIA-7B	.0611	.0973	.1041	.0453	.0770
	instruct	Minerva-7B	.0419	.0498	.0520	.0260	.0424
		EuroLLM-9B	.0238	.1143	.1233	.0260	.0719
		Velvet-14B	.0294	.0317	.0317	.0272	.0300
		FastwebMIIA-7B	.0396	.0848	.0815	.0441	.0625
		LLaVA-NeXT 8B	.0022	**.1810**	**.1810**	**.1176**	**.1205**
EXAMS-V-IT	base	Minerva-7B	.1655	.2117	.0000	.0658	.1108
		EuroLLM-9B	.2420	.2402	.2367	.2331	.2380
		FastwebMIIA-7B	.2438	**.2580**	.2402	**.2456**	**.2469**
	instruct	Minerva-7B	**.2456**	.2456	.0000	.0260	.1293
		EuroLLM-9B	.2420	.2402	.2402	.2384	.2402
		Velvet-14B	.1833	.1744	.1673	.2420	.1918
		FastwebMIIA-7B	2438	.2438	**.2438**	.2438	.2438
		LLaVA-NeXT 8B	.0000	.2171	.1299	.0160	.0908

nodes. The software stack was based on open-source libraries, including Transformers from Hugging Face [19], which provides seamless integration with PyTorch [20] and DeepSpeed [21]. This software stack has been instrumental in efficiently handling large data sets and complex models.

This hardware-software configuration ensured reproducibility, scalability and efficiency, which are crucial for the comparative analysis of multiple model architectures and for training large-scale models on Italian-language data. It also reflects a broader national effort towards a sovereign AI infrastructure, ensuring data localisation, transparency and regulatory compliance.

For training the models, we use 2 GPUs. The whole training procedure takes about 24 hours for each model.

4. Experiments

4.1. Experimental Setting

To evaluate the vision-language ability of these models, we use three datasets: GQA-IT [22, 23], MTVQA [24], EXAMS-V [25]. GQA-IT is a visual question answering dataset on natural scenes. We consider its manually translated split to Italian, consisting of 3,000 instances. MTVQA is a manually annotated text-centric image dataset. The dataset provides splits for several languages, in this work we focus on the Italian split, which consists of 884 question-answer pairs. We refer to it as MTVQA-IT. EXAMS-V is a collection of multiple-choice school exam questions in multiple languages. In this case,

Table 3
We report results for GQA-IT and MTVQA-IT for the approximate match setting. Specifically, we report the formatting where the models performed **worst** in the original setting and compare the two results (exact and approximate).

Dataset	Type	Model	Formatting	Exact Match	Approximate Match
GQA-IT	base	Minerva-7B	Post-Swap	.2023	.4133
		EuroLLM-9B	Post-Swap	.0973	.4807
		FastwebMIIA-7B	Post-Swap	.0297	.4853
	instruct	Minerva-7B	Post-Swap	.0533	.4610
		EuroLLM-9B	Pre	.0370	.4977
		Velvet-14B	Post-Swap	.2843	.2967
		FastwebMIIA-7B	Post-Swap	.0790	.4846
		LLaVA-NeXT 8B	Pre	.0009	.4709
MTVQA-IT	base	Minerva-7B	Post-Swap	.0373	.0656
		EuroLLM-9B	Post-Swap	.0792	.1358
		FastwebMIIA-7B	Post-Swap	.0453	.1301
	instruct	Minerva-7B	Post-Swap	.0260	.0724
		EuroLLM-9B	Pre	.0238	.1652
		Velvet-14B	Post-Swap	.0272	.0305
		FastwebMIIA-7B	Pre	.0396	.1244
		LLaVA-NeXT 8B	Pre	.0022	.2398

we focus on the Italian split as well, which consists of 1,645 question-answer pairs. We refer to it as EXAMS-V-IT

To take into account the effect of using different prompts for the same model, we test all models and all datasets using four different styles of formatting. Specifically, to evaluate these models, an additional string is added to the prompt to limit the generated output. In English, this string that is used depends on the model and the formatting of its training mixture, however the original LLaVA, and most other models following its setup, used "Answer the question using a single word or phrase." for open-ended tasks and "Answer with the option's letter from the given choices directly." for closed-ended ones. Thanks to this, it is possible to use *exact match* as metric, where the generated output is compared directly to the ground truth (i.e. hard syntactic match), since the model is instructed to generate only the text that is relevant w.r.t. the label. Due to this, we want to understand if and how the model performance is affected by this string. If we change this string to one with a similar meaning, does the model generate outputs consistently? Does the position of the string matter? To answer these questions, we apply four different formattings to the datasets:

- **Pre**: "Rispondi in modo breve e diretto.\s" (or "Rispondi con la lettera.\s" for closed-ended tasks) appended to the **beginning** of the instruction
- **Post**: "\nRispondi utilizzando una sola parola o frase." (or "\nRispondi utilizzando direttamente la lettera dell'opzione corretta tra quelle date." for closed-ended tasks) appended to the **end** of the instruction
- **Pre-Swap**: "Rispondi utilizzando una sola parola o frase.\n" (or "Rispondi utilizzando direttamente la lettera dell'opzione corretta tra quelle date.\n" for closed-ended tasks) appended to the **beginning** of the instruction
- **Post-Swap**: "\sRispondi in modo breve e diretto." (or "\sRispondi con la lettera." for closed-ended tasks) appended to the **end** of the instruction

A model that performs well for all four formattings can be considered to be a consistent model, capable of answering user queries despite the syntax used in the request. Finally, all results are obtained using greedy decoding as sampling strategy at inference time, which removes randomness in generation and guarantees improved reproducibility of the obtained results. For all tasks, we use the question and answer pairs provided by the task itself. The only exception is EXAMS-V-IT where, since the question and choices are embedded within the image itself, we use the following string as question: "Fornisci una risposta alla domanda presente nell'immagine." ("Provide an answer to the question in the image" in English). All

Table 4
We report results for GQA-IT and MTVQA-IT for the approximate match setting. Specifically, we report the formatting where the models performed **best** in the original setting and compare the two results (exact and approximate).

Dataset	Type	Model	Formatting	Exact Match	Approximate Match
GQA-IT	base	Minerva-7B	Post	.3523	.3723
		EuroLLM-9B	Pre-Swap	.4157	.4497
		FastwebMIIA-7B	Post	.4147	.4313
	instruct	Minerva-7B	Post	.3520	.3640
		EuroLLM-9B	Pre-Swap	.4187	.4283
		Velvet-14B	Pre-Swap	.3107	.3147
		FastwebMIIA-7B	Post	.3233	.3543
		LLaVA-NeXT 8B	Pre-Swap	.3454	.3520
MTVQA-IT	base	Minerva-7B	Post	.0577	.0634
		EuroLLM-9B	Pre-Swap	.1143	.1290
		FastwebMIIA-7B	Pre-Swap	.1041	.1165
	instruct	Minerva-7B	Pre-Swap	.0520	.0656
		EuroLLM-9B	Pre-Swap	.1233	.1324
		Velvet-14B	Post	.0317	.0362
		FastwebMIIA-7B	Post	.0848	.0928
		LLaVA-NeXT 8B	Pre-Swap	.1810	.1991

models are evaluated using the *lmms-eval*[7] framework, loaded in float16 as dtype and inference is performed with a batch size of 1, ensuring reproducibility of the results. Finally, we lowercase text and ground truth and ignore whitespaces when evaluating using *exact match*.

4.2. Results Discussion

We report the results of the experiments in Table 2. For the sake of comparison against already existing models, we also report the results of LLaVA-NeXT 8B [26], a LLaVA-NeXT model trained from the LLaMA 3 Instruct 8B checkpoint, on these benchmarks. Overall, models trained on Italian perform well w.r.t. LLaVA-NeXT 8B. Remarkably, the base version of EuroLLM has the best average performance in GQA, while the base version of FastwebMIIA has the best average performance in EXAMS-V-IT. In MTVQA-IT, Italian models tend to perform poorly w.r.t. LLaVA-NeXT 8B. We believe this is due to the low quantity of text-centric vision-language instances in the training mixture, since MultiInstruct tasks focus more on natural scenes and everyday images. We can reasonably expect an improvement in performance for text-centric tasks when integrating this type of tasks in the training mixture.

Additionally, we showcase that the models are very sensitive to the formatting of the prompt. For example, while the base version of EuroLLM achieves the best average performance on GQA-IT, it performs well on only two out of the four formattings. This pattern can also be seen in other models in our evaluation, in most cases, the models tend to perform better in a limited subset of formattings. After manually analyzing the generated outputs, we find that there are cases where the models generated the correct answer, but with additional contextual text. For example, for the question "È nuvoloso?" ("Is it cloudy" in English) with label "Sì" ("Yes" in English), Minerva instruct answered "Sì" in the Post formatting, while it answered with "Sì, è nuvoloso nell'immagine." ("Yes, it is cloudy in the image" in English) in the Post-Swap formatting. In both cases, the answer is correct, but the *exact match* metric fails to consider the second case as correct, since there is no hard syntactic match between the generated output and the label. In light of this, we propose further evaluation to study the relationship between performance and the length of the generated response.

4.3. Evaluating for Response Length

To further understand if the models provide outputs that are relevant, we evaluate them by performing an approx-

[7] https://github.com/EvolvingLMMs-Lab/lmms-eval

LABEL	auto. *car.*	parcheggio. *parking lot.*
PRE FORMATTING	Un'auto sta aspettando il semaforo. *A car is waiting for the traffic light.*	P è per il parcheggio. *P is for the parking lot.*
POST FORMATTING	auto. *car.*	parcheggio. *parking lot.*

Figure 1: Visualization of some examples from GQA-IT and MTVQA-IT for the problem of evaluating using *exact match* as metric. For both formattings (Pre and Post in this example) the model correctly generates the output response, however the *exact match* metric fails to capture the correctness of the response for the Pre formatting. Beneath each Italian text we provide its corresponding English translation.

imate match between the label and the generated output. That is, we check that the label is a substring of the generated output. This allows us to cover cases where the model keeps generating contextual text together with the task answer. For example, for the question "C'è una palla da calcio nell'immagine?" ("Is there a football ball in the image?" in English) with label "Sì" ("Yes" in English), the model may generate "Sì, c'è una palla da calcio nell'immagine.". This case is considered incorrect in the *exact match* metric, since the generated output is not the same as the ground truth label. However, the answer is correct, and the ground truth label is in the generated string itself. Our approach allows to cover these corner cases, however, note that this strategy suffers from false positives. For example, for the question "C'è una mano nell'immagine?" ("Is there a hand in the image?" In English) with label "No", the model may generate "Sì, c'è una mano nell'immagine" ("Yes, there is a hand in the image" in English), and it would be considered a correct answer since "no" is a substring of "mano". We showcase some examples in Figure 1 To assess the performance of the models regardless of the response length, we consider the formatting where each model has performed the worst. We retrieve the generated outputs and corresponding ground truth labels and evaluate them using an approximate match. We expect an improvement in performance w.r.t. *exact match*. Note that we do not perform this evaluation for EXAMS-V, since the task is closed-ended, the answers are the identifiers of the options (e.g. "A", "B"), making it impossible to evaluate the task using this strategy. Results for evaluation performed using this approach are reported in Table 3. As expected, we can appreciate a great improvement in performance for most models. For example, for the base version of EuroLLM-9B, performance rises from .0973 to .4807, and a similar trend can be seen in the instruct version of the model. For most models, we can observe an increase in performance in approximate match, except for Velvet, where the performance remains the same. To further validate this finding, we also evaluate under the same setting the formatting where the models performed best, Results for approximate match evaluation of the best formatting are reported in Table 4. Overall, the results are a lot more stable, and the degree of improvement is less with respect to worse formatting using approximate match. This highlights that the models in their best formatting performed well because they were able to generate the expected output directly and consistently, without adding additional contextual text to the answer. However, we emphasize that the worst formatting evaluated with approximate match actually showcases better performance w.r.t. best formatting evaluated with approximate match. For example, the base version of EuroLLM achieves an approximate match of .4807 on GQA-IT for its worst formatting, while it achieves an approximate match of .4497 for its best one. This pattern can be seen for all models, including LLaVA NeXT, the only exception being Velvet, where performance is consistent for both formattings. This finding highlights that LLMs tend to provide better answers when they are able to provide a

Table 5
We report zero-shot results for Global-MMLU-Lite on the Italian language for each subset. For each model and each category of the dataset, we underline the best result between the multimodal model (LVLM) and its original checkpoint (LLM).

Type	Model	Multimodal	Global MMLU Subset					
			Business	Humanities	Medical	Others	Social Sciences	Stem
base	Minerva-7B	X	.3103	.3431	.4167	.4107	.3235	.4130
	Minerva-7B	✓	.4310	.3725	.4444	.5000	.3137	.3478
	EuroLLM-9B	X	.6207	.6176	.3889	.6786	.5392	.3478
	EuroLLM-9B	✓	.5862	.5980	.6111	.6786	.6471	.5000
	FastwebMIIA-7B	X	.2931	.3824	.4444	.4643	.3137	.3261
	FastwebMIIA-7B	✓	.4655	.4216	.6111	.5357	.4020	.2609
instruct	Minerva-7B	X	.2931	.3922	.4167	.4286	.3529	.3478
	Minerva-7B	✓	.3103	.3725	.4722	.3393	.2549	.2391
	EuroLLM-9B	X	.5862	.6667	.5556	.6964	.5784	.3913
	EuroLLM-9B	✓	.6379	.6275	.6944	.6429	.5882	.4783
	Velvet-14B	X	.5345	.6176	.7222	.6607	.6569	.5870
	Velvet-14B	✓	.3448	.3039	.4167	.4107	.3039	.2609
	FastwebMIIA-7B	X	.5345	.6373	.5833	.6250	.6569	.5217
	FastwebMIIA-7B	✓	.5172	.5490	.6111	.6786	.5784	.4130

verbalized response.

4.4. Evaluating for Text-only Tasks

Finally, we also test the ability of the LVLMs in solving Italian text-only tasks, rather than vision-language ones. This aims to determine whether the models retain the knowledge they learned during their original text-only training procedure. Since the models didn't see text-only data during vision-language training, we expect their performance to be lower with respect to their original LLM version. Since we only want to have a general estimate of their performance, we consider a relatively small subset of Italian tasks available through the *lm-eval-harness*[8] framework. Namely, we consider *Global-MMLU* [27], specifically its LITE subset. The dataset is a balanced collection of culturally sensitive and culturally agnostic MMLU tasks (a massive multitask test dataset consisting of multiple-choice questions from various branches of knowledge), where only languages with human translation and post-edits are included. Results are reported in Table 5. Surprisingly, there are models which perform better after the visual instruction-tuning step. For example, the base version of Minerva-7B performs better on four out of the six categories of the dataset. Similar behaviour is also showcased by other models, for example, the instruct version of EuroLLM-9B also performs better on four out of the six categories, while the base version of FastwebMIIA performs better on five of them. This showcases that a vision-language training procedure may also enhance the language-only performance of the model. However, there is an outlier to this pattern, that is Velvet-14B, where the original version of the model performs better on all categories. Furthermore, for the other models, there is no consistent improvement across all categories. This highlights that, while multimodality has helped improve the inherent knowledge of these models, it is not guaranteed, and text-only evaluation is still relevant for multimodal models.

[8] https://github.com/EleutherAI/lm-evaluation-harness

5. Conclusions

In this work, we have expanded the current landscape of LVLMs for the Italian language. We have collected a pool of LLMs supporting the Italian language, which only process textual inputs. Then, we have extended them to LVLMs, by employing a state-of-the-art approach, namely LLaVA-NeXT, and a machine-translated corpus of vision-language tasks in Italian. Additionally, we evaluated them using only benchmarks that are natively in Italian and also studied the effect on the length of the generated response in evaluation. Finally, we also benchmarked these models on an Italian text-only benchmark to understand if the performance for text-only tasks was worse after the visual instruction-tuning step. As future work, we plan to further extend the training mixture so that it also considers text-centric tasks in Italian, improving model performance on this type of task that is currently missing in the training mixture. Specifically, we plan to incorporate multimodal document data to enhance these models in document visual question an-

swering. We also plan to further extend the evaluation and to improve the approximate match strategy, which soundness currently suffers from the possibility of false positives.

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

[1] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al., Llama 2: Open foundation and fine-tuned chat models, arXiv preprint arXiv:2307.09288 (2023).

[2] R. Orlando, L. Moroni, P.-L. H. Cabot, S. Conia, E. Barba, S. Orlandini, G. Fiameni, R. Navigli, Minerva llms: The first family of large language models trained from scratch on italian data, in: Proceedings of the 10th Italian Conference on Computational Linguistics (CLiC-it 2024), 2024, pp. 707–719.

[3] H. Liu, C. Li, Q. Wu, Y. J. Lee, Visual instruction tuning, Advances in neural information processing systems 36 (2023) 34892–34916.

[4] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: Transformers for image recognition at scale, arXiv preprint arXiv:2010.11929 (2020).

[5] H. Liu, C. Li, Y. Li, Y. J. Lee, Improved baselines with visual instruction tuning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 26296–26306.

[6] B. Li, Y. Zhang, D. Guo, R. Zhang, F. Li, H. Zhang, K. Zhang, P. Zhang, Y. Li, Z. Liu, et al., Llava-onevision: Easy visual task transfer, arXiv preprint arXiv:2408.03326 (2024).

[7] J. Li, D. Li, S. Savarese, S. Hoi, Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models, in: International conference on machine learning, PMLR, 2023, pp. 19730–19742.

[8] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, et al., Flamingo: a visual language model for few-shot learning, Advances in neural information processing systems 35 (2022) 23716–23736.

[9] P. H. Martins, P. Fernandes, J. Alves, N. M. Guerreiro, R. Rei, D. M. Alves, J. Pombal, A. Farajian, M. Faysse, M. Klimaszewski, et al., Eurollm: Multilingual language models for europe, arXiv preprint arXiv:2409.16235 (2024).

[10] E. Musacchio, L. Siciliani, P. Basile, G. Semeraro, Llava-ndino: Empowering llms with multimodality for the italian language, in: Proceedings of the Eighth Workshop on Natural Language for Artificial Intelligence (NL4AI 2024) co-located with 23th International Conference of the Italian Association for Artificial Intelligence (AI* IA 2024), CEUR-WS. org, 2024.

[11] A. Bacciu, C. Campagnano, G. Trappolini, F. Silvestri, DanteLLM: Let's push Italian LLM research forward!, in: N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti, N. Xue (Eds.), Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), ELRA and ICCL, Torino, Italia, 2024, pp. 4343–4355. URL: https://aclanthology.org/2024.lrec-main.388/.

[12] L. Moroni, S. Conia, F. Martelli, R. Navigli, et al., Itabench: Towards a more comprehensive evaluation for italian llms, in: Proceedings of the Tenth Italian Conference on Computational Linguistics (CLiC-it 2024), 2024.

[13] G. Attanasio, P. Basile, F. Borazio, D. Croce, M. Francis, J. Gili, E. Musacchio, M. Nissim, V. Patti, M. Rinaldi, et al., Calamita: Challenge the abilities of language models in italian, in: Proceedings of the 10th Italian Conference on Computational Linguistics (CLiC-it 2024), Pisa, Italy, 2024.

[14] X. Wang, C. Hu, B. Ma, P. Röttger, B. Plank, Look at the text: Instruction-tuned language models are more robust multiple choice selectors than you think, arXiv preprint arXiv:2404.08382 (2024).

[15] C.-H. Chiang, H.-y. Lee, Can large language models be an alternative to human evaluations?, in: A. Rogers, J. Boyd-Graber, N. Okazaki (Eds.), Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Toronto, Canada, 2023, pp. 15607–15631. URL: https://aclanthology.org/2023.acl-long.870/. doi:10.18653/v1/2023.acl-long.870.

[16] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, I. Sutskever, Learning transferable visual models from natural language supervision, 2021. URL: https://arxiv.org/abs/2103.00020. arXiv:2103.00020.

[17] S. Kudugunta, I. Caswell, B. Zhang, X. Garcia, D. Xin, A. Kusupati, R. Stella, A. Bapna, O. Firat, Madlad-400: A multilingual and document-level

large audited dataset, Advances in Neural Information Processing Systems 36 (2023) 67284–67296.

[18] Z. Xu, Y. Shen, L. Huang, Multiinstruct: Improving multi-modal zero-shot learning via instruction tuning, in: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2023, pp. 11445–11465.

[19] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, A. M. Rush, Transformers: State-of-the-art natural language processing, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Online, 2020, pp. 38–45. URL: https://www.aclweb.org/anthology/2020.emnlp-demos.6.

[20] J. Ansel, E. Yang, H. He, N. Gimelshein, A. Jain, M. Voznesensky, B. Bao, P. Bell, D. Berard, E. Burovski, G. Chauhan, A. Chourdia, W. Constable, A. Desmaison, Z. DeVito, E. Ellison, W. Feng, J. Gong, M. Gschwind, B. Hirsh, S. Huang, K. Kalambarkar, L. Kirsch, M. Lazos, M. Lezcano, Y. Liang, J. Liang, Y. Lu, C. Luk, B. Maher, Y. Pan, C. Puhrsch, M. Reso, M. Saroufim, M. Y. Siraichi, H. Suk, M. Suo, P. Tillet, E. Wang, X. Wang, W. Wen, S. Zhang, X. Zhao, K. Zhou, R. Zou, A. Mathews, G. Chanan, P. Wu, S. Chintala, Pytorch 2: Faster machine learning through dynamic python bytecode transformation and graph compilation, in: 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS '24), ACM, 2024. URL: https://pytorch.org/assets/pytorch2-2.pdf. doi:10.1145/3620665.3640366.

[21] C. Li, Z. Yao, X. Wu, M. Zhang, C. Holmes, C. Li, Y. He, Deepspeed data efficiency: Improving deep learning model quality and training efficiency via efficient data sampling and routing, 2024. URL: https://arxiv.org/abs/2212.03597. arXiv:2212.03597.

[22] D. A. Hudson, C. D. Manning, Gqa: A new dataset for real-world visual reasoning and compositional question answering, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 6700–6709.

[23] D. Croce, L. C. Passaro, A. Lenci, R. Basili, et al., Gqa-it: Italian question answering on image scene graphs, in: Italian Conference on Computational Linguistics 2021 Proceedings of the Eighth Italian Conference on Computational Linguistics, volume 3033, 2021.

[24] J. Tang, Q. Liu, Y. Ye, J. Lu, S. Wei, C. Lin, W. Li, M. F. F. B. Mahmood, H. Feng, Z. Zhao, et al., Mtvqa: Benchmarking multilingual text-centric visual question answering, arXiv preprint arXiv:2405.11985 (2024).

[25] R. J. Das, S. E. Hristov, H. Li, D. I. Dimitrov, I. Koychev, P. Nakov, Exams-v: A multi-discipline multilingual multimodal exam benchmark for evaluating vision language models, arXiv preprint arXiv:2403.10378 (2024).

[26] H. Liu, C. Li, Y. Li, B. Li, Y. Zhang, S. Shen, Y. J. Lee, Llava-next: Improved reasoning, ocr, and world knowledge, 2024. URL: https://llava-vl.github.io/blog/2024-01-30-llava-next/.

[27] S. Singh, A. Romanou, C. Fourrier, D. I. Adelani, J. G. Ngui, D. Vila-Suero, P. Limkonchotiwat, K. Marchisio, W. Q. Leong, Y. Susanto, R. Ng, S. Longpre, W.-Y. Ko, M. Smith, A. Bosselut, A. Oh, A. F. T. Martins, L. Choshen, D. Ippolito, E. Ferrante, M. Fadaee, B. Ermis, S. Hooker, Global mmlu: Understanding and addressing cultural and linguistic biases in multilingual evaluation, 2024. URL: https://arxiv.org/abs/2412.03304. arXiv:2412.03304.