Training Small LLMs from Scratch: Lessons from Zagreus and Nesso

Alessandro Ercolani[1], Edoardo Federici[1], Luisa Bentivogli[2] and Luca Corbucci[2]

[1]*mii-llm, Via Guglielmi 2/E, 38057, Pergine Valsugana (TN), Italy*
[2]*Fondazione Bruno Kessler, MT Unit, Via Sommarive 18, 38123, Povo (TN), Italy*

Abstract

Small Language Models (SLMs) are increasingly attractive for deployment on consumer hardware, edge devices, and privacy-sensitive applications, yet most open efforts remain centered on English and a small set of high-resource languages. This paper introduces **Zagreus** and **Nesso**, two closely related families of openly released SLMs designed for the Romance-language ecosystem, as part of an ongoing research effort. We report here on the first release of sub-billion-parameter models; larger variants and extended evaluations are in active development. Zagreus consists of four bilingual foundation models pairing English with Italian, Spanish, Portuguese, and French, respectively; each model has approximately 400 million parameters and is trained from scratch on roughly one trillion tokens. Nesso extends these foundations through supervised fine-tuning for instruction following and agentic function calling, including a fully open variant trained only on publicly available data. Beyond model release, we describe the full training pipeline, including data curation, tokenizer design, architecture choices, distributed pre-training, post-training, and evaluation. We assess the models on localized MMLU, HellaSwag, ARC, IFEval, Portuguese leaderboard tasks, and the Italian EVALITA suite. The results show that carefully curated data and transparent training practices allow sub-billion-parameter models to remain competitive with larger baselines of comparable class: `Nesso-0.4B-instruct` matches or exceeds `Qwen3-0.6B` on several Italian and English benchmarks, while `Zagreus-0.4B-por` outperforms `Qwen3-0.6B-Base` on the Portuguese leaderboard evaluation. These findings indicate that language-focused SLMs can provide practical, reproducible alternatives to larger general models, while highlighting structured prediction and discriminative classification as important targets for future data augmentation and alignment work.

Keywords

Small Language Models, European Languages, Pre-training, Post-training, Evaluation, Reproducibility

1. Introduction

Recent years have witnessed the rapid emergence and proliferation of Large Language Models (LLMs) [1, 2, 3]. These models have evolved into sophisticated virtual assistants, demonstrating advanced capabilities in natural language understanding, machine translation [4], and code generation [5]. Furthermore, LLMs are increasingly integrated into agentic frameworks to assist human operators across a diverse array of complex tasks [6, 7]. However, a fundamental limitation of these models is their scale: the same computational demands that define them as "large" make deployment on constrained hardware prohibitive, requiring high-performance GPUs for efficient deployment and inference. This computational bottleneck, coupled with the growing demand for multi-agent systems capable of securely interacting with localized data and executing device-level operations, has catalyzed a new research paradigm. Consequently, focus has shifted toward developing highly capable, energy-efficient architectures designed for deployment on consumer hardware and edge environments, giving rise to the domain of Small Language Models (SLMs) [8, 9]. Despite the proliferation of SLM research, the vast majority of existing efforts remain concentrated on English [8, 9] and a small number of high-resource languages. European languages in general, and Romance languages in particular, remain underserved, with few openly available models trained from scratch on dedicated multilingual corpora [10, 11, 12, 13, 14, 15, 16]. This disparity carries practical consequences: models trained predominantly on English data exhibit degraded performance on Italian, Spanish, Portuguese, and French, both in terms of linguistic fluency

CLiC-it 2026: Twelfth Italian Conference on Computational Linguistics, September 14–16, 2026, Palermo, Italy
ale.ercolani@gmail.com (A. Ercolani); edoardo.federici@outlook.com (E. Federici); bentivo@fbk.eu (L. Bentivogli); lcorbucci@fbk.eu (L. Corbucci)
© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

and cultural grounding. The absence of transparent, reproducible baselines further compounds the problem, making it difficult for the research community to build upon or critically evaluate existing work. In this paper, we address these gaps through the **mii-llm** initiative[1], a fully open and reproducible effort to build competitive SLMs for the Romance language ecosystem from first principles.

We train and release an initial subset of the **Zagreus** family of bilingual foundation models, covering English–Italian, English–Spanish, English–Portuguese, and English–French. Each model comprises approximately 400 million parameters and is trained on roughly 1 trillion tokens. These models represent the first generation of the Zagreus family; work is actively ongoing to extend pre-training, broaden benchmark coverage, and scale to larger model variants. Building on these base models, we further develop and release the **Nesso** family of post-trained variants. All three models, `Nesso-0.4B-instruct`, `Nesso-0.4B-agentic`, and `Open-Zagreus-0.4B`, are obtained via supervised fine-tuning (SFT), which adapts a pre-trained base model to follow instructions by optimising a standard cross-entropy loss over target tokens in curated prompt-response pairs [17]. `Nesso-0.4B-instruct` and `Nesso-0.4B-agentic` use `Zagreus-0.4B-ita` as their base model and are trained on a bilingual English–Italian corpus: the former targets general-purpose instruction-following and conversational use cases, while the latter is specialised for structured output generation, function calling, and agentic execution. `Open-Zagreus-0.4B` is a fully open variant trained exclusively on publicly available data. These capabilities are especially relevant for on-device and edge deployment, where compact, low-latency models offer practical advantages over larger alternatives. At present, post-training has been completed for the English–Italian variant only; supervised fine-tuning for the remaining language pairs is ongoing and will be released as part of this continuing effort. We summarize our key contributions below:

- We present a complete, reproducible pipeline for training SLMs from scratch, covering data curation and tokenization, distributed pre-training, and post-training via Supervised Fine-Tuning (SFT) [17].
- We release four bilingual foundation models and three post-trained variants, reporting preliminary evaluations on standard multilingual benchmarks including MMLU [18], HellaSwag [19], ARC [20], IFEval [21], and the Italian EVALITA suite [22].
- We demonstrate that carefully engineered small models can achieve competitive performance against state-of-the-art models of comparable scale.
- We document our data sources, framework choices, and training configurations openly to ensure reproducible research.

Our models are publicly available on Hugging Face, our code is open source and available on GitHub, and part of the training data is publicly accessible through Hugging Face datasets, including FineWeb sample-350BT [23], FineWeb-2 Italian, FinePDFs Italian, and StarCoderData.

2. Related Works

2.1. Small Language Models

The aim of improving the performance of LLMs has led companies and researchers to produce large models with an ever-increasing number of parameters [1, 24, 2]. On one side, these models offer frontier performance on a huge variety of tasks, from coding [25], to reasoning [26], to RAG [27]. On the other side, they have strict hardware requirements that only a few companies can afford. In most cases, these frontier models are developed by companies that tend not to share details of how they trained their models. This makes it more difficult to replicate the training and to achieve similar performance.

Although the recent release of diverse open-weight and open-source models, together with inference frameworks, has made local execution easier, interest is growing in models that are small enough to

[1] https://www.mii-llm.ai/

run on consumer devices [28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38]. Such models can perform specific tasks efficiently while supporting languages typically overlooked by larger models. Moreover, while LLMs offer strong performance, their reliance on centralised cloud infrastructure introduces challenges regarding scalability, environmental impact [39, 40], and privacy [41, 42]. This explains the interest in the field of Small Language Models (SLMs) [8, 43, 44, 45], defined in the recent literature as decoder-only transformer models typically ranging from 100M to 5B parameters [9]. SLMs are purposefully designed for deployment on edge devices and consumer hardware with limited computational resources [45, 9]. A recurring finding across this literature is that data quality and curation strategy are as important as raw parameter count in the small-parameter regime [46, 47]. English-centric SLMs illustrate this well. **TinyLlama** [48] (1.1B) demonstrated that a LLaMA-2-style architecture trained on up to 3 trillion tokens can outperform similarly sized open models. **MobileLLM** [49] introduced a deep-and-thin design with grouped query attention (GQA) and block-wise weight sharing, targeting sub-1B on-device deployment and achieving accuracy gains over prior sub-1B models. **SmolLM2** [47] (1.7B) was overtrained on approximately 11 trillion tokens drawn from FineWeb-Edu, DCLM, FineMath, and The Stack, using a multi-stage curriculum to progressively sharpen mathematical and coding capabilities, ultimately surpassing contemporaries such as `Qwen2.5-1.5B` [50] and `Llama-3.2-1B` [51] on standard benchmarks. **Phi-3 and -4-mini** [52, 53] (3.8B) showed that synthetic, curriculum-driven pretraining data can yield strong reasoning and instruction-following at a fraction of typical parameter budgets, with the Phi-4-mini variant extending coverage to a 200K multilingual vocabulary and achieving performance on par with models twice its size on mathematical benchmarks. **Gemma 2, 3 and 4** [54, 28, 55] further pushed the efficiency frontier. Gemma 3 extended this with multimodal input while Gemma 4 introduced Per-Layer Embeddings (PLE) to further reduce active computation per token on edge hardware, with the E2B and E4B variants achieving effective 2.3B and 4.5B active parameter footprints under full multimodal input. Beyond English, **Qwen2.5** [50] (0.5B–3B) provides multilingual coverage across approximately 29 languages with up to 18 trillion pretraining tokens, while **Llama 3.2** [51] (1B, 3B) offers officially supported Romance language coverage, French, Italian, Portuguese, and Spanish, through pruning and knowledge distillation from Llama 3.1 8B.

2.2. Multilingual and Romance-Language SLMs

Within the multilingual domain, two families provide the broadest European coverage at the SLM scale. **EuroLLM** [11] (1.7B) was pretrained on 4 trillion tokens spanning all 24 official EU languages plus 11 strategic languages, with increased high-quality data emphasis in the final 10% of training. **Salamandra-2B** [15] extended this to 35 European languages, including Spanish, Catalan, Italian, French, Portuguese, Romanian, Occitan, and Galician, training from scratch on 12 trillion tokens of curated multilingual data. **Falcon 3** [56] (1B, 3B) covers all four target EuroRomance languages and was obtained through pruning and distillation from Falcon3-7B, itself trained from scratch on 14 trillion tokens. Openly released SLMs specifically tailored for *bilingual* Romance–English pairings at sub-4B scale, however, remain rare. **CroissantLLM** [10] (1.3B) is the closest methodological analogue in the French–English space: it applies a strict 1:1 French-to-English token ratio across 3 trillion tokens with a custom bilingual tokenizer and introduces FrenchBench as a dedicated evaluation suite. For Italian–English bilingualism, **Minerva** [16, 57] (350M, 1B, 3B, 7B) represents the most direct prior work, training from scratch on exact 50/50 Italian–English token splits using a Mistral-style architecture with RoPE, GQA, and SwiGLU. **Velvet-2B** [58] offers a recent industrial Italian–English alternative trained on over 2 trillion tokens with a 127K-vocabulary tokenizer and 32K context length. For Portuguese, **Tucano** [14] (160M–2.4B) provides a systematic family of PT-BR-only models trained on 200B tokens, while **GlórIA** [59] (1.3B, 2.7B) covers European Portuguese. Our Nesso models follow the bilingual recipe established by Minerva and CroissantLLM, extending it to the English–Italian pair at the sub-4B scale while incorporating the post-training alignment pipeline described below.

2.3. Post-Training and Alignment

Post-training alignment is now a standard component of the LLM training pipeline [17]. Supervised Fine-Tuning (SFT) on curated instruction datasets is typically followed by Direct Preference Optimization (DPO) [60], which aligns model outputs with human preferences without requiring an explicit reward model. Recent SLMs have additionally integrated *thinking* modes that allow small architectures to perform multi-step chain-of-thought reasoning, a capability previously associated only with much larger systems [3, 61]. Data curation for post-training follows the same quality-over-quantity principle as pre-training. SmolLM2 applies SFT to SmolTalk and DPO to the UltraFeedback preference dataset [62], while OLMo [35] uses the fully open Tülu-3 SFT and DPO pipeline [37].

Our Nesso models follow this paradigm, applying SFT on curated bilingual Italian–English instruction datasets derived from the Zagreus base.

3. Architecture

The Zagreus and Nesso model families are built on a modified, fully dense Llama-3.2 transformer architecture [51, 63]. This choice is motivated by three main considerations. First, the Llama architecture is among the most widely adopted open-weight model families, with extensive support across inference engines, fine-tuning frameworks, and hardware backends [51, 64]. Second, Llama-3.2 is supported directly by Nanotron[2], the Hugging Face pre-training library we use. This support eliminates the need for custom architecture adapters and reduces engineering overhead. Third, dense transformer architectures offer more predictable compute use and more stable training dynamics at the SLM scale [51, 65]. These properties are important for models with relatively few parameters and limited compute budgets. By contrast, sparse Mixture-of-Experts architectures [66, 67] introduce routing overhead, reducing the efficiency advantages in the case of SLMs.

Both Zagreus and Nesso adopt the design choices of the Llama-3.2 family [51]. The models use pre-layer normalization through RMSNorm [68], SwiGLU activations in the feed-forward sublayers [69], Grouped Query Attention (GQA) for memory-efficient inference [70], and Rotary Position Embeddings (RoPE) for relative positional encoding [71]. RMSNorm stabilizes the activations, SwiGLU provides nonlinear transformation, GQA reduces memory required by attention, and RoPE represents token positions relative to one another. These choices support stable optimization and efficient inference.

We use the `Llama-3.2-1B` tokenizer, which uses Byte-Pair Encoding (BPE) [72] with a vocabulary of 128,256 tokens. BPE represents text as frequent subword units, allowing the same vocabulary to cover words and word fragments across languages. We select this tokenizer for its multilingual coverage across English, Italian, Spanish, Portuguese, and French. A shared vocabulary across the four Zagreus variants also ensures consistent tokenization and enables direct cross-lingual comparison.

The main architectural hyperparameters are reported in Table 1.

4. Training

4.1. Pre-Training

Pre-training is the most demanding stage of the pipeline. Each of the four Zagreus base models is trained on approximately one trillion tokens, balancing model capacity with available compute budget.

Training framework. We use Nanotron, a minimalist distributed pre-training library for transformer models that natively supports 3D parallelism (data, tensor, and pipeline). Its integration with the Hugging Face ecosystem, including Accelerate and Datasets, keeps the pipeline cohesive across all stages, from data tokenization to checkpoint release. To better suit our infrastructure requirements, we maintained an actively developed fork of Nanotron[3], incorporating patches and optimisations

[2] https://github.com/huggingface/nanotron
[3] https://github.com/mii-llm/nanotron

Table 1
Main architectural and training hyperparameters of the `Zagreus-0.4B` Base models.

Hyperparameter	Value	Hyperparameter	Value
Model		*Optimizer*	
Parameters (approx.)	400M	Algorithm	AdamW (fused)
Architecture	Dense transformer (Llama-3.2 style)	β_1, β_2 / ϵ	$(0.9,\ 0.95)$ / 10^{-8}
		Weight decay	0.01
Dimensions		Gradient clipping	1.0
Hidden layers	32	*Learning rate schedule*	
Hidden size	960	Schedule	Linear warm-up + decay
Feed-forward size	2,560	Peak/min. LR	3×10^{-3} / 10^{-7}
Attention		Warm-up steps	4,000
Attention heads	15	Decay start step	750,000
Key-value heads	5	Decay steps	50,000
Mechanism	GQA		
Activation & norm.		*Batch configuration*	
Activation function	SwiGLU / SiLU	Global batch size	256
Normalization	Pre-layer RMSNorm ($\epsilon = 10^{-5}$)	Micro-batch size	4
Context & position		*Tokenizer*	
Context length	4,096 tokens	Tokenizer	Llama-3.2 BPE
Positional encoding	RoPE ($\theta = 10{,}000$)	Vocabulary size	128,256
		Tied embeddings	Yes
		Initializer range	0.02

specifically targeting SLURM-managed HPC clusters.

Tokenization and data preparation. We perform data preprocessing and tokenization using DataTrove[4], a scalable pipeline library developed by Hugging Face for large-scale text processing. Preparing the full one-trillion-token corpus required over three weeks of continuous computation on 32 CPU cores. For each bilingual model, the token distribution comprises approximately 400B tokens of English text and 400B tokens of the respective target language (Italian, Spanish, Portuguese, or French), following findings on effective multilingual data mixtures [73, 74]. The remaining 200B tokens consist of source code drawn from the StarCoder dataset [75], selected in light of evidence that using code during pre-training improves structured reasoning and instruction-following capabilities in LLMs [76, 77].

Training configuration. We train the models with a standard autoregressive language-modeling objective: the model predicts each token from the preceding tokens. We optimize the objective with AdamW [78], using linear warm-up followed by linear learning-rate decay. We monitor loss curves and run intermediate benchmark evaluations to track capability growth and detect training instability or catastrophic forgetting. The main pre-training configuration is summarized in Table 1. Each optimization step uses a micro-batch of 4, no gradient accumulation beyond the local micro-batch, and data parallelism over 64 GPUs. This gives an effective global batch size of 256 per step.

Compute and infrastructure. We run all pre-training experiments on the Seeweb HPC cluster[5], utilising 64 NVIDIA A100 GPUs distributed across multiple nodes for almost seven days. Job scheduling and fault recovery were managed via SLURM.

[4] https://github.com/huggingface/datatrove
[5] https://www.seeweb.it/

4.2. Post-Training

Post-training turns a pre-trained base model into a deployable assistant [17]. It requires less compute and less data than pre-training, but the data must be carefully curated because its quality largely determines the resulting gains.

Framework. We conduct all post-training experiments using Axolotl [6], a flexible fine-tuning framework that abstracts over low-level distributed training details and supports multiple training paradigms, including supervised fine-tuning (SFT) [17], and direct preference optimisation (DPO) [60]. We integrated this framework into our SLURM-based HPC infrastructure to support distributed multi-GPU execution and maximize parallelism during post-training.

Supervised Fine-Tuning. All the released post-trained models, `Nesso-0.4B-instruct`, `Nesso-0.4B-agentic`, and `Open-Zagreus-0.4B`, are at the SFT stage. SFT adapts a pre-trained base model to follow instructions by optimising a standard cross-entropy loss over target tokens in curated prompt-response pairs [17]. `Nesso-0.4B-instruct` and `Nesso-0.4B-agentic` use `Zagreus-0.4B-ita` as their base model and are trained on a bilingual English–Italian corpus. `Nesso-0.4B-instruct` targets general-purpose instruction following and conversational use cases, whereas `Nesso-0.4B-agentic` is specialised for structured output generation, function calling, and agentic execution. These capabilities are especially relevant for on-device and edge deployment, where compact, low-latency models offer practical advantages over larger alternatives.

The fine-tuning dataset used for both `Nesso-0.4B-instruct` and `Nesso-0.4B-agentic` comprises approximately 2.5M English and Italian samples collected over several years. The samples span a broad range of domains to promote generalisation across diverse instruction types. As this dataset constitutes a proprietary research asset, we do not release it publicly. However, we release the model weights on Hugging Face[7] alongside a comprehensive, fully reproducible evaluation based on open-source libraries.

By contrast, we fine-tune `Open-Zagreus-0.4B` exclusively on publicly available data, specifically, the *OpenItalianData* dataset[8] containing approximately 2 million synthetically generated conversations in Italian obtained using state-of-the-art models. This model serves as a fully reproducible, open-source baseline, demonstrating that competitive bilingual English–Italian SLMs can be developed end-to-end even without reliance on proprietary data.

5. Evaluation

This section presents a systematic evaluation of our trained models. First, we report the results for the Zagreus foundation models in Section 5.1. We then report the results for the post-trained models in Section 5.2. Due to space constraints, we report and analyse here only the results obtained with the base Italian model Zagreus-0.4B and on the post-trained models `Nesso-0.4B-instruct`, `Nesso-0.4B-agentic`, and `Zagreus-0.4B-open` on the Italian language. The results obtained with `Zagreus-0.4B` for Spanish, Portuguese, and French are reported in Section A in the Appendix. These results are overall similar and coherent with the ones obtained with the Italian model.

5.1. Zagreus Base Model Evaluation

The evaluation of the Zagreus Base model allows us to assess whether the data curation strategy and architectural design produce models that acquire competitive multilingual representations despite their

[6] https://axolotl.ai/
[7] https://huggingface.co/collections/mii-llm/zagreus-nesso-fine-tuned
[8] https://huggingface.co/datasets/DeepMount00/OpenItalianData

Table 2
Comparison between Zagreus Base model (0.4B parameters) and other models of similar size. We report MMLU-IT accuracy (5-shot setting), HellaSwag-IT `acc_norm`, and ARC-IT `acc_norm` (0-shot setting). Results of our Zagreus Italian model are comparable with those of Minerva 1B and `LFM2-350M`.

Model	MMLU_{it}	$\text{HellaSwag}_{\text{it}}$	ARC_{it}	Average
Zagreus-0.4B-ita-base	0.253	0.375	0.258	0.296
Minerva 350M	0.254	0.314	0.241	0.269
Minerva 1B	0.251	0.382	0.267	0.300
Minerva 3B	0.261	0.518	0.304	0.361
LFM2-350M	0.313	0.299	0.246	0.286
Qwen3-0.6B-Base	0.431	0.394	0.310	0.378

small parameter budget. All experiments are conducted using lm-evaluation-harness[9], to which we contribute with multilingual benchmark support.

Each model is evaluated on three widely adopted benchmarks:

- **MMLU** [18]: a broad-coverage academic benchmark spanning 57 subjects, measuring world knowledge and multitasking reasoning.
- **HellaSwag** [19]: a common sense natural language inference benchmark requiring models to complete activity descriptions.
- **ARC** [20]: a science question answering benchmark targeting elementary-level reasoning.

During pre-training, each bilingual base model is evaluated only on the localized benchmark variant of its target Romance language. Therefore, for `Zagreus-0.4B-ita-base`, we report Italian MMLU, HellaSwag, and ARC results; English evaluations are considered in the post-training evaluation of the Nesso models. Table 2 compares the scores obtained with the final checkpoint of `Zagreus-0.4B-ita-base` with other base models. We compare our model with a part of the Minerva family (350M, 1B, and 3B parameters), with LFM2 from LiquidAI (350M parameters), and with Qwen 3 (600M parameters). We chose to compare with Minerva because, similarly to our base model, it is pretrained from scratch on Italian. Moreover, we compare with LFM2 and Qwen 3 because of their similar parameter size. For the Minerva models, the Italian benchmark is already available in the "Open Ita LLM Leaderboard" maintained by mii-llm[10]. For Qwen and LFM2 instead, we run the benchmarks from scratch. The results obtained with the Zagreus base model trained on Italian are comparable with the ones obtained with Minerva 1B and better compared with Minerva 350M, demonstrating the effectiveness of Zagreus pre-training. Overall, Qwen is the model that obtains the best scores, while LFM2 is, on average, comparable with Zagreus.

To assess Italian-language capabilities beyond general-purpose multilingual benchmarks, we also evaluate `Zagreus-0.4B-ita-base` on the EVALITA benchmark suite [22]. The results are reported and analysed in 5.2.1, where we compare the base model with `Open-Zagreus-0.4B`.

To characterize training dynamics and support reproducibility, we report in Appendix A all the other benchmark results obtained during the training of the base models.

5.2. Post-Trained Nesso Models Evaluation

This section evaluates the instruction-tuned Nesso model family: `Nesso-0.4B-instruct` and `Nesso-0.4B-agentic`. Both models are derived from `Zagreus-0.4B-ita-base` and post-trained via supervised fine-tuning (SFT). Since the underlying pre-training is bilingual (English–Italian), evaluations are performed in both languages.

[9] https://github.com/EleutherAI/lm-evaluation-harness
[10] https://huggingface.co/spaces/mii-llm/open_ita_llm_leaderboard

Table 3
English and Italian benchmark results for post-trained Nesso models and baselines. Metrics: IFEval (prompt-level accuracy), ARC (`acc_norm`), HellaSwag (`acc_norm`), and MMLU (5-shot accuracy). Cell shading indicates per-column ranking (darker = higher score).

Model	$\text{IFEval}_{\text{EN}}$	ARC_{EN}	HS_{EN}	MMLU_{EN}	Avg_{EN}	$\text{IFEval}_{\text{IT}}$	ARC_{IT}	HS_{IT}	MMLU_{IT}	Avg_{IT}	Overall
Nesso-0.4B-instruct	0.3465	0.3003	0.4629	0.2871	0.3492	0.2962	0.2874	0.4076	0.2875	0.3197	0.3345
Nesso-0.4B-agentic	0.2962	0.2534	0.4062	0.2889	0.3112	0.2914	0.2541	0.3673	0.2730	0.2965	0.3039
Qwen3-0.6B	0.2758	0.3430	0.4742	0.4013	0.3736	0.3058	0.2729	0.3598	0.4025	0.3353	0.3545
LFM2-350M	0.1595	0.2457	0.3092	0.3445	0.2647	0.1427	0.2464	0.2994	0.3132	0.2504	0.2576

In addition to MMLU, ARC, and HellaSwag, in this phase, we also include **IFEval** [21], a benchmark that evaluates a model's ability to follow verifiable natural language instructions, measuring compliance with explicit constraints such as output format, length, and content requirements.

In Table 3 we report results obtained comparing `Nesso-0.4B-instruct` and `Nesso-0.4B-agentic` against two externally available baselines of comparable scale: `Qwen3-0.6B` [29] and `LFM2-350M` [79].

`Qwen3-0.6B` achieves the highest MMLU scores in both languages. This can reflect either a broader multilingual knowledge coverage at scale or the inclusion of MMLU-style data in large-scale training pipelines, a phenomenon increasingly noted in the literature [80]. However, `Nesso-0.4B-instruct` outperforms Qwen on IFEval in English and scores comparably on HellaSwag, demonstrating that our post-training procedure meaningfully improves instruction-following and commonsense reasoning while remaining competitive across other dimensions. Both Nesso variants substantially outperform `LFM2-350M` across all benchmarks and both languages, confirming the effectiveness of the full pre-training and post-training pipeline. These results collectively support the feasibility of building high-quality bilingual small language models optimized for edge inference in resource-constrained English–Italian deployment scenarios.

5.2.1. Open-Zagreus-0.4B

`Open-Zagreus-0.4B-ita` is a fully open-source variant of the Nesso family, trained exclusively on publicly available Italian data[11]. This experiment is motivated by reproducibility and accessibility: we aim to assess whether competitive English–Italian models can be constructed entirely from open datasets and open training pipelines, without reliance on proprietary data sources.

Table 4 reports Italian benchmark results; in this case, we compare the results with post-trained models of similar size evaluated in the "Open Ita LLM Leaderboard". Specifically, we selected the SmolLM2 models (360M and 1.7B parameters) developed by HuggingFace, `Falcon3-1B`, and `Gemma-3-1B`. Moreover, we compare the results of Open-Zagreus with our two post-trained models to understand the impact of the data quality on the final results. The results obtained with Open-Zagreus are comparable to those obtained with Falcon and better than those obtained with SmolLM2-360M. The scores obtained by `Nesso-0.4B-instruct` on these benchmarks are slightly better than those obtained with Open-Zagreus, likely due to the proprietary post-training dataset used in the former case. Compared with `Nesso-0.4B-agentic`, instead, the results are comparable, with a bigger difference in the ARC_{it} benchmark. Overall, these results align with our expectations, demonstrating that post-training on high-quality, openly available data can yield competitive bilingual models without reliance on proprietary resources.

EVALITA Comparison with Base Model: To characterize the effect of supervised fine-tuning on Italian task performance, we compare `Open-Zagreus-0.4B` against the base `Zagreus-0.4B-ita` on the EVALITA benchmark suite [22]. EVALITA provides a comprehensive set of Italian NLP tasks spanning classification, sequence labeling, semantic similarity, summarization, and reasoning, offering a more fine-grained view of language-specific competencies.

[11] https://huggingface.co/datasets/DeepMount00/OpenItalianData

Table 4
Italian benchmark results for `Open-Zagreus-0.4B` and `Nesso-0.4B` models compared to instruct baselines of similar size. We report MMLU-IT accuracy (5-shot), HellaSwag-IT, and ARC-IT `acc_norm` (0-shot). Cell shading indicates per-column ranking (darker = higher score).

Model	MMLU_{it}	$\text{HellaSwag}_{\text{it}}$	ARC_{it}	Average
Open-Zagreus-0.4B	0.253	0.361	0.302	0.305
Nesso-0.4B-instruct	0.287	0.407	0.287	0.319
Nesso-0.4B-agentic	0.273	0.367	0.254	0.296
SmolLM2-360M-Instruct	0.264	0.267	0.304	0.278
SmolLM2-1.7B-Instruct	0.347	0.376	0.276	0.335
Falcon3-1B-Instruct	0.341	0.321	0.262	0.308
Gemma-3-1B	0.364	0.474	0.349	0.396

Table 5
Per-task EVALITA [22] results for `Zagreus-0.4B-ita-base` and `Open-Zagreus-0.4B`. **Bold** indicates the best result per task. Δ is Open-Zagreus minus base.

Task (Metric)	Zagreus-0.4B-ita-base	Open-Zagreus-0.4B	Δ
Gains after fine-tuning			
EVALITA-LLM (acc)	0.3226	**0.3313**	+0.0087
Sentiment Analysis (F1)	0.3660	**0.3712**	+0.0052
Summarization (ROUGE-1)	0.1947	**0.2305**	+0.0358
Text Entailment (acc)	0.5133	**0.5492**	+0.0359
Word-in-Context (F1)	0.4697	**0.4880**	+0.0183
Regressions after fine-tuning			
Admission Test (acc)	**0.2137**	0.2083	−0.0054
FAQ (acc)	**0.2681**	0.2672	−0.0009
Hate Speech Detection (F1)	**0.6056**	0.4340	−0.1716
NER (F1)	**0.1611**	0.1357	−0.0254
Relation Extraction (F1)	**0.1244**	0.0437	−0.0807

Table 5 reports per-task results. The base model achieves its strongest scores on hate-speech detection, textual entailment, and word-in-context. Structured prediction tasks, relation extraction, and named entity recognition remain challenging at this scale, consistent with prior work on compact language models [81].

Fine-tuning on open Italian data yields a net positive effect on generative and semantic tasks, with the most notable gains on text entailment, summarization, and word-in-context. However, performance regresses markedly on structured prediction tasks, hate-speech detection, and relation extraction, suggesting that the open dataset underrepresents the annotation styles required by these tasks. The Δ column makes this trade-off immediately visible: fine-tuning consistently improves the top half of the table while degrading the bottom. This pattern is consistent with a known tension in instruction tuning, where gains in generative and semantic tasks may come at the cost of discriminative structured-prediction capabilities [82, 83].

Collectively, these results confirm that competitive bilingual small language models can be developed using fully open-source resources, while also identifying specific task categories, structured prediction, and discriminative classification, where targeted data augmentation would yield the greatest returns.

6. Conclusion

This work studied how small language models can be scientifically meaningful and practically useful when their design is guided by careful data selection, transparent engineering, and targeted evaluation. The Zagreus and Nesso families show that training compact models from first principles is not only feasible but also valuable for deployment scenarios where efficiency, reproducibility, and linguistic coverage matter as much as aggregate benchmark performance.

6.1. Key Contributions

In this work, we have presented the **Zagreus** and **Nesso** model families, a suite of seven openly released Small Language Models built from first principles and optimized for the Romance language ecosystem. Zagreus comprises four bilingual foundation models, pairing English with Italian, Spanish, Portuguese, and French, respectively, each with approximately 400 million parameters trained on roughly 1 trillion tokens. Nesso extends these base models through Supervised Fine-Tuning, yielding two post-trained variants optimized for conversational instruction-following and agentic tool-calling, alongside a fully open-source variant trained exclusively on publicly available data.

Our results demonstrate that carefully engineered small models can achieve competitive performance against state-of-the-art models of comparable scale across standard multilingual benchmarks, including MMLU [18], HellaSwag [19], ARC [20], IFEval [21], and the Italian EVALITA suite [22]. In particular, `Nesso-0.4B-instruct` performs competitively with `Qwen3-0.6B` on several benchmarks despite having fewer parameters, and `Zagreus-0.4B-por` shows strong performance relative to `Qwen3-0.6B-Base` on the Portuguese leaderboard evaluation. These results validate our central hypothesis: that focused data curation, sound architectural choices, and a reproducible training pipeline can yield competitive performance within the sub-billion parameter regime.

Beyond the empirical results, a core contribution of this paper is its transparency. We have documented every stage of the development pipeline, motivation, framework selection, data engineering, tokenization, pre-training, and post-training, with sufficient detail to allow independent replication. We release all model checkpoints publicly on Hugging Face, and the mii-llm community releases the full training codebase on GitHub. The `Open-Zagreus-0.4B` model additionally demonstrates that competitive English–Italian SLMs can be built end-to-end using exclusively open-source data.

6.2. Lessons Learned

Separate Bilingual Models. The initial goal of the project was to train a single English–Italian model. After validating the English–Italian recipe, we kept the English data fixed and changed only the Romance-language data, whose availability and comparability enabled replication. This turns separate bilingual releases into matched experiments, allowing future work to measure transfer and interference across languages without changing the entire training recipe. The value of this design is therefore not only specialization, but also comparability.

Reproducibility. Training from scratch is a long engineering process. Releasing the data sources, configurations, checkpoints, and weights lets the community test which parts of the recipe matter instead of treating the final model as an isolated artifact. At the time of writing, members of the mii-llm community are already extending the released models through fine-tuning and additional post-training.

Post-Training. Post-training improves instruction following but not every capability equally. Table 5 shows gains on semantic and generative tasks alongside losses on structured and discriminative tasks. The lesson here is that SFT data must preserve the structured and discriminative behaviors required by the evaluation, not only conversational quality.

Scale the Data Before the Architecture. With approximately 1T tokens for a 0.4B model, the training recipe obtains competitive results using a conventional dense architecture. The practical lesson is not that more tokens always solve the problem, but that data scale and curation deserve priority before architectural complexity. Because the experiments do not disentangle token count, data quality, and code data, this conclusion remains a hypothesis for controlled follow-up experiments.

6.3. Limitations and Future Work

Several directions remain open for future work. First, the Nesso models are currently at the SFT stage; completing the DPO [60] alignment step is expected to yield further gains in instruction adherence and response quality, and updated results will be released accordingly. This ongoing experiment, performed via the mii-llm community data pipeline [12], represents an opportunity to further close the gap with proprietary frontier models at this parameter scale. Second, while our bilingual models cover four Romance languages, a natural extension is a unified multilingual model trained jointly on all four target languages, which could improve cross-lingual transfer and reduce the total number of models required for deployment. Third, our evaluation focused on classification and generation benchmarks; a deeper investigation of long-form generation quality, factual grounding, and safety properties in the target languages would strengthen confidence in the models for real-world deployment.

We emphasise that this paper represents a preliminary report on an actively ongoing research effort. Pre-training for the English–Spanish, English–Portuguese, and English–French variants is continuing, with updated checkpoints to be released upon completion. In parallel, we are developing larger-scale model variants to extend the Zagreus and Nesso families to a broader set of capacities and use cases.

We hope that the Zagreus and Nesso families, together with this paper, serve as a useful resource for the broader community working on efficient multilingual language models, and that the transparency of our approach contributes to reducing technological and geopolitical dependencies in AI research.

Declaration on Generative AI

During the preparation of this work, the author(s) used AI tools for grammar and spelling refinement (ChatGPT and Grammarly). After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

Acknowledgments

We gratefully acknowledge Seeweb (seeweb.it) for providing the computing resources used in this work. We also thank Michele Montebovi (deepmount) for his contribution to the open SFT datasets used for post-training.

This paper has received funding from the European Union's Digital Europe Programme (DIGITAL) under grant agreement 101198470, project LLMs4EU.

References

[1] T. Brown et al., Language models are few-shot learners, in: H. Larochelle et al. (Eds.), Advances in Neural Information Processing Systems, volume 33, Curran Associates, Inc., 2020, pp. 1877–1901.

[2] G. Comanici et al., Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities, 2025. URL: https://arxiv.org/abs/2507.06261. arXiv:2507.06261.

[3] DeepSeek-AI et al., Deepseek-v3 technical report, 2025. URL: https://arxiv.org/abs/2412.19437. arXiv:2412.19437.

[12] https://www.mii-llm.ai/

[4] M. Finkelstein et al., Translategemma technical report, 2026. URL: https://arxiv.org/abs/2601.09012. arXiv:2601.09012.
[5] J. Liu et al., Large language model-based agents for software engineering: A survey, 2025. URL: https://arxiv.org/abs/2409.02977. arXiv:2409.02977.
[6] Y. Talebirad et al., Multi-agent collaboration: Harnessing the power of intelligent llm agents, arXiv preprint arXiv:2306.03314 (2023).
[7] Y. Li et al., Personal llm agents: Insights and survey about the capability, efficiency and security, arXiv preprint arXiv:2401.05459 (2024).
[8] F. Wang et al., A comprehensive survey of small language models in the era of large language models: Techniques, enhancements, applications, collaboration with llms, and trustworthiness, ACM Transactions on Intelligent Systems and Technology 16 (2025) 1–87.
[9] Z. Lu et al., Small language models: Survey, measurements, and insights, 2025. URL: https://arxiv.org/abs/2409.15790. arXiv:2409.15790.
[10] M. Faysse et al., Croissantllm: A truly bilingual french-english language model, Transactions on Machine Learning Research (2024).
[11] P. H. Martins et al., Eurollm: Multilingual language models for europe, Procedia Computer Science 255 (2025) 53–62. URL: https://www.sciencedirect.com/science/article/pii/S1877050925006210. doi:https://doi.org/10.1016/j.procs.2025.02.260, proceedings of the Second EuroHPC user day.
[12] P. H. Martins et al., Eurollm-9b: Technical report, 2025. URL: https://arxiv.org/abs/2506.04079. arXiv:2506.04079.
[13] M. M. Ramos et al., Eurollm-22b: Technical report, 2026. URL: https://arxiv.org/abs/2602.05879. arXiv:2602.05879.
[14] N. K. Corrêa et al., Tucano: Advancing neural text generation for portuguese, Patterns 6 (2025) 101325. URL: http://dx.doi.org/10.1016/j.patter.2025.101325. doi:10.1016/j.patter.2025.101325.
[15] A. Gonzalez-Agirre et al., Salamandra technical report, 2025. URL: https://arxiv.org/abs/2502.08489. arXiv:2502.08489.
[16] R. Orlando et al., Minerva LLMs: The first family of large language models trained from scratch on Italian data, in: F. Dell'Orletta et al. (Eds.), Proceedings of the Tenth Italian Conference on Computational Linguistics (CLiC-it 2024), CEUR Workshop Proceedings, Pisa, Italy, 2024, pp. 707–719. URL: https://aclanthology.org/2024.clicit-1.77/.
[17] L. Ouyang et al., Training language models to follow instructions with human feedback, in: Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22, Curran Associates Inc., Red Hook, NY, USA, 2022.
[18] D. Hendrycks et al., Measuring massive multitask language understanding, in: International Conference on Learning Representations, 2021. URL: https://openreview.net/forum?id=d7KBjmI3GmQ.
[19] R. Zellers et al., HellaSwag: Can a Machine Really Finish Your Sentence?, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL), 2019.
[20] P. Clark et al., Think you have solved question answering? Try ARC, the AI2 Reasoning Challenge, arXiv preprint arXiv:1803.05457 (2018).
[21] J. Zhou et al., Instruction-following evaluation for large language models, 2023. arXiv:2311.07911.
[22] M. Lai et al., EVALITA 2023: Overview of the 8th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian, in: Proceedings of the Eighth Evaluation Campaign of NLP and Speech Tools for Italian (EVALITA 2023), CEUR.org, 2023.
[23] G. Penedo et al., Fineweb2: One pipeline to scale them all — adapting pre-training data processing to every language, in: Second Conference on Language Modeling, 2025. URL: https://openreview.net/forum?id=jnRBe6zatP.
[24] G. Team et al., Gemini: A family of highly capable multimodal models, 2025. URL: https://arxiv.org/abs/2312.11805. arXiv:2312.11805.
[25] Y. Dong et al., A survey on code generation with llm-based agents, arXiv preprint arXiv:2508.00083

(2025).
[26] D. Guo et al., Deepseek-r1 incentivizes reasoning in llms through reinforcement learning, Nature 645 (2025) 633–638. URL: http://dx.doi.org/10.1038/s41586-025-09422-z. doi:10.1038/s41586-025-09422-z.
[27] M. Arslan et al., A survey on rag with llms, Procedia computer science 246 (2024) 3781–3790.
[28] G. Team et al., Gemma 3 technical report, 2025. URL: https://arxiv.org/abs/2503.19786. arXiv:2503.19786.
[29] A. Yang et al., Qwen3 technical report, 2025. URL: https://arxiv.org/abs/2505.09388. arXiv:2505.09388.
[30] Q. Team, Qwen3.5-omni technical report, 2026. URL: https://arxiv.org/abs/2604.15804. arXiv:2604.15804.
[31] K. Team et al., Kimi k2.5: Visual agentic intelligence, 2026. URL: https://arxiv.org/abs/2602.02276. arXiv:2602.02276.
[32] K. Team et al., Kimi k2: Open agentic intelligence, 2026. URL: https://arxiv.org/abs/2507.20534. arXiv:2507.20534.
[33] NVIDIA et al., Nvidia nemotron 3: Efficient and open intelligence, 2025. URL: https://arxiv.org/abs/2512.20856. arXiv:2512.20856.
[34] . Team et al., Glm-4.5: Agentic, reasoning, and coding (arc) foundation models, 2025. URL: https://arxiv.org/abs/2508.06471. arXiv:2508.06471.
[35] D. Groeneveld et al., OLMo: Accelerating the science of language models, in: L.-W. Ku et al. (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 15789–15809. URL: https://aclanthology.org/2024.acl-long.841/. doi:10.18653/v1/2024.acl-long.841.
[36] T. Olmo et al., Olmo 3, 2026. URL: https://arxiv.org/abs/2512.13961. arXiv:2512.13961.
[37] N. Lambert et al., Tulu 3: Pushing frontiers in open language model post-training, in: Second Conference on Language Modeling, 2025. URL: https://openreview.net/forum?id=i1uGbfHHpH.
[38] W. Kwon et al., Efficient memory management for large language model serving with pagedattention, in: Proceedings of the 29th Symposium on Operating Systems Principles, SOSP '23, Association for Computing Machinery, New York, NY, USA, 2023, p. 611–626. URL: https://doi.org/10.1145/3600006.3613165. doi:10.1145/3600006.3613165.
[39] J. Fernandez et al., Energy considerations of large language model inference and efficiency optimizations, in: W. Che et al. (Eds.), Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Vienna, Austria, 2025, pp. 32556–32569. URL: https://aclanthology.org/2025.acl-long.1563/. doi:10.18653/v1/2025.acl-long.1563.
[40] J. Stojkovic et al., Towards greener llms: Bringing energy-efficiency to the forefront of llm inference, 2024. URL: https://arxiv.org/abs/2403.20306. arXiv:2403.20306.
[41] B. Yan et al., On protecting the data privacy of large language models (llms) and llm agents: A literature review, High-Confidence Computing 5 (2025) 100300. URL: https://www.sciencedirect.com/science/article/pii/S2667295225000042. doi:https://doi.org/10.1016/j.hcc.2025.100300.
[42] Y. Du et al., Beyond data privacy: New privacy risks for large language models, 2026. URL: https://arxiv.org/abs/2509.14278. arXiv:2509.14278.
[43] F. Corradini et al., State of the art and future directions of small language models: a systematic review, Big Data and Cognitive Computing 9 (2025) 189.
[44] J. Xu et al., On-device language models: A comprehensive review, 2024. URL: https://arxiv.org/abs/2409.00088. arXiv:2409.00088.
[45] P. Belcak et al., Small language models are the future of agentic ai, 2025. URL: https://arxiv.org/abs/2506.02153. arXiv:2506.02153.
[46] G. Penedo et al., The fineweb datasets: Decanting the web for the finest text data at scale, Advances in Neural Information Processing Systems 37 (2024) 30811–30849.

[47] L. B. Allal et al., Smollm2: When smol goes big – data-centric training of a small language model, 2025. URL: https://arxiv.org/abs/2502.02737. arXiv:2502.02737.
[48] P. Zhang et al., Tinyllama: An open-source small language model, 2024. URL: https://arxiv.org/abs/2401.02385. arXiv:2401.02385.
[49] Z. Liu et al., Mobilellm: Optimizing sub-billion parameter language models for on-device use cases, in: Forty-first International Conference on Machine Learning, 2024.
[50] Qwen et al., Qwen2.5 technical report, 2025. URL: https://arxiv.org/abs/2412.15115. arXiv:2412.15115.
[51] A. Grattafiori et al., The llama 3 herd of models, arXiv preprint arXiv:2407.21783 (2024).
[52] M. Abdin et al., Phi-3 technical report: A highly capable language model locally on your phone, 2024. URL: https://arxiv.org/abs/2404.14219. arXiv:2404.14219.
[53] M. Abdin et al., Phi-4 technical report, 2024. URL: https://arxiv.org/abs/2412.08905. arXiv:2412.08905.
[54] G. Team et al., Gemma 2: Improving open language models at a practical size, 2024. URL: https://arxiv.org/abs/2408.00118. arXiv:2408.00118.
[55] Google DeepMind, Gemma 4: Byte for byte, the most capable open models, https://blog.google/innovation-and-ai/technology/developers-tools/gemma-4/, 2026.
[56] F.-L. Team, The falcon 3 family of open models, 2024. URL: https://huggingface.co/blog/falcon3.
[57] L. Moroni et al., What we learned from continually training minerva: A case study on Italian, in: C. Bosco et al. (Eds.), Proceedings of the Eleventh Italian Conference on Computational Linguistics (CLiC-it 2025), CEUR Workshop Proceedings, Cagliari, Italy, 2025, pp. 760–774. URL: https://aclanthology.org/2025.clicit-1.72/.
[58] Almawave, Velvet-2b: Italian–english bilingual language model, https://huggingface.co/almawave/Velvet-2B, 2025.
[59] R. Lopes et al., Glória: A generative and open large language model for portuguese, in: Proceedings of the 16th International Conference on Computational Processing of Portuguese-Vol. 1, 2024, pp. 441–453.
[60] R. Rafailov et al., Direct Preference Optimization: Your Language Model is Secretly a Reward Model, in: Advances in Neural Information Processing Systems (NeurIPS), 2023.
[61] OpenAI et al., Openai o1 system card, 2026. URL: https://arxiv.org/abs/2412.16720. arXiv:2412.16720.
[62] G. Cui et al., Ultrafeedback: boosting language models with scaled ai feedback, in: Proceedings of the 41st International Conference on Machine Learning, ICML'24, JMLR.org, 2024.
[63] H. Touvron et al., Llama: Open and efficient foundation language models, 2023. URL: https://arxiv.org/abs/2302.13971. arXiv:2302.13971.
[64] Y. Zheng et al., Llamafactory: Unified efficient fine-tuning of 100+ language models, in: Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 3: system demonstrations), 2024, pp. 400–410.
[65] B. Pan et al., Dense training, sparse inference: Rethinking training of mixture-of-experts language models, arXiv preprint arXiv:2404.05567 (2024).
[66] W. Fedus et al., Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity, Journal of Machine Learning Research 23 (2022) 1–39.
[67] N. Shazeer et al., Outrageously large neural networks: The sparsely-gated mixture-of-experts layer, in: International Conference on Learning Representations, 2017. URL: https://openreview.net/forum?id=B1ckMDqlg.
[68] B. Zhang et al., Root mean square layer normalization, Advances in neural information processing systems 32 (2019).
[69] N. Shazeer, Glu variants improve transformer, 2020. URL: https://arxiv.org/abs/2002.05202. arXiv:2002.05202.
[70] J. Ainslie et al., Gqa: Training generalized multi-query transformer models from multi-head checkpoints, in: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, 2023, pp. 4895–4901.

[71] J. Su et al., Roformer: Enhanced transformer with rotary position embedding, 2024. URL: https://doi.org/10.1016/j.neucom.2023.127063. doi:10.1016/j.neucom.2023.127063.
[72] R. Sennrich et al., Neural machine translation of rare words with subword units, in: K. Erk et al. (Eds.), Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Berlin, Germany, 2016, pp. 1715–1725. URL: https://aclanthology.org/P16-1162/. doi:10.18653/v1/P16-1162.
[73] N. Foroutan et al., Revisiting multilingual data mixtures in language model pretraining, 2025. URL: https://arxiv.org/abs/2510.25947. arXiv:2510.25947.
[74] Z. Li et al., Rethinking multilingual continual pretraining: Data mixing for adapting LLMs across languages and resources, in: Second Conference on Language Modeling, 2025. URL: https://openreview.net/forum?id=mpTIzK4Zca.
[75] R. Li et al., Starcoder: may the source be with you!, Transactions on Machine Learning Research 2023 (2023). Publisher Copyright: © 2023, Transactions on Machine Learning Research. All rights reserved.
[76] V. Aryabumi et al., To code or not to code? exploring impact of code in pre-training, in: International Conference on Learning Representations, volume 2025, 2025, pp. 79469–79495.
[77] Y. MA et al., At which training stage does code data help LLMs reasoning?, in: The Twelfth International Conference on Learning Representations, 2024.
[78] I. Loshchilov et al., Decoupled weight decay regularization, in: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019, OpenReview.net, 2019.
[79] A. Amini et al., Lfm2 technical report, 2025. URL: https://arxiv.org/abs/2511.23404. arXiv:2511.23404.
[80] K. Zhou et al., Don't make your LLM an evaluation benchmark cheater, CoRR abs/2311.01964 (2023). URL: https://doi.org/10.48550/arXiv.2311.01964. doi:10.48550/ARXIV.2311.01964. arXiv:2311.01964.
[81] I. Turc et al., Well-read students learn better: On the importance of pre-training compact models, 2019. URL: https://arxiv.org/abs/1908.08962. arXiv:1908.08962.
[82] S. Ghosh et al., A closer look at the limitations of instruction tuning, in: Proceedings of the 41st International Conference on Machine Learning, 2024.
[83] S. Longpre et al., The flan collection: Designing data and methods for effective instruction tuning, in: International conference on machine learning, PMLR, 2023, pp. 22631–22648.

A. Additional Experiments

Here, we report the additional experiments and evaluations performed on the other Zagreus base models. Specifically, we present results for the English–French, English–Spanish, and English–Portuguese variants, which are released at the current pre-training checkpoint. Pre-training is still ongoing for these models; benchmark evaluation and supervised fine-tuning are planned as subsequent steps once training is complete.

A.1. English–Spanish Zagreus 0.4B Base

Table 6 reports evaluation results for the Spanish foundational model. Consistent with the Italian variant, MMLU scores remain stable throughout training while HellaSwag and ARC improve progressively, suggesting that the bilingual pre-training objective supports steady acquisition of commonsense reasoning capabilities in Spanish.

A.2. English–French Zagreus 0.4B Base

Table 7 reports results for the French foundational model across training steps. The pattern of early MMLU stabilization followed by gradual gains on reasoning-oriented benchmarks mirrors observations from the Italian and Spanish variants, suggesting this dynamic is robust across the Romance languages considered.

A.3. English–Portuguese Zagreus 0.4B Base

Table 8 reports results for the Portuguese foundational model. Among all language variants evaluated, the Portuguese model achieves the strongest aggregate performance, with consistent improvements across checkpoints—most notably on HellaSwag (0.3732 $\rightarrow$ 0.3865) and ARC (0.2667 $\rightarrow$ 0.2786).

To situate the Portuguese model within a broader comparative context, we evaluate it on the benchmark suite introduced by "Lm-Evaluation-Harness-PT"[13] alongside publicly available models on the associated open leaderboard[14]. This suite spans a diverse set of Portuguese NLP tasks, including textual entailment, semantic similarity, legal reasoning, offensive language detection, and sentiment analysis, providing a more comprehensive assessment of Portuguese language competence than the standard MMLU/ARC/HellaSwag triplet alone.

Table 9 compares two Zagreus checkpoints against `Qwen3-0.6B-Base`. The Zagreus 483k checkpoint achieves the highest overall average (0.3230), outperforming both the later Zagreus 582k checkpoint and the Qwen3 baseline (0.2569), despite the latter having a larger parameter count.

A.4. English–Italian Zagreus 0.4B Base

[13] https://github.com/eduagarcia/lm-evaluation-harness-pt
[14] https://huggingface.co/spaces/eduagarcia/open_pt_llm_leaderboard

Table 6
Checkpoint progression for Zagreus-0.4B Base trained on English–Spanish dataset. For each checkpoint, we report MMLU (5-shot accuracy), HellaSwag (acc_norm), and ARC (acc_norm).

Steps	$MMLU_{es}$	ARC_{es}	$HellaSwag_{es}$	Average
146k	0.254	0.265	0.409	0.309
216k	0.237	0.270	0.414	0.307
292k	0.254	0.262	0.417	0.311
406k	0.254	0.269	0.423	0.315
518k	0.255	0.280	0.429	0.321

Table 7
Checkpoint progression for Zagreus-0.4B Base trained on English–French dataset. For each checkpoint, we report MMLU (5-shot accuracy), HellaSwag (acc_norm), and ARC (acc_norm).

Steps	$MMLU_{fr}$	ARC_{fr}	$HellaSwag_{fr}$	Average
129k	0.262	—	—	0.262
231k	0.263	—	—	0.263
365k	0.256	0.278	0.414	0.316
456k	0.267	—	—	0.267
603k	0.256	0.280	0.415	0.318
705k	0.266	0.281	0.417	0.321

Table 8
Benchmark results across checkpoints for Zagreus-0.4B-por-Base.

Checkpoint	ARC_{pt}	$HellaSwag_{pt}$	$MMLU_{pt}$	Average
153k	0.2667	0.3732	0.2685	0.3028
207k	0.2705	0.3768	0.2671	0.3048
276k	0.2718	0.3789	0.2664	0.3057
345k	0.2564	0.3796	0.2669	0.3009
414k	0.2682	0.3842	0.2673	0.3066
483k	0.2667	0.3865	0.2658	0.3063
582k	0.2786	0.3865	0.2688	0.3113

Table 9
Portuguese leaderboard evaluation. Tasks: RTE (textual entailment), STS (semantic textual similarity), BLUEX (university entrance exam), ENEM (national high-school exam), FAQUAD NLI (legal NLI), HateBR (hate speech), OAB (bar exam), PT Hate (offensive language), TweetSent (sentiment). Cell shading indicates per-column ranking (darker = higher score).

Model	RTE	STS	BLUEX	ENEM	FAQUAD NLI	HateBR	OAB	PT Hate	TweetSent	**Average**
Zagreus 483k	0.4624	0.1650	0.2434	0.2071	0.4397	0.3327	0.2528	0.4817	0.3220	**0.3230**
Zagreus 582k	0.3361	0.0449	0.2100	0.1903	0.4397	0.3825	0.2392	0.4444	0.1542	0.2713
Qwen3-0.6B-Base	0.3333	0.0726	0.1057	0.0077	0.4397	0.3333	0.0428	0.4123	0.5646	0.2569

Table 10
Checkpoint progression for Zagreus-0.4B Base trained on English–Italian dataset. For each checkpoint, we report MMLU (5-shot accuracy), HellaSwag (`acc_norm`), and ARC (`acc_norm`).

Checkpoint	MMLU_{it}	$\text{HellaSwag}_{\text{it}}$	ARC_{it}	Average
v2-95k	0.2529	0.3366	0.2652	0.2849
v2-205k	0.2628	—	—	0.2628
v2-290k	0.2428	0.3492	0.2335	0.2752
v2-305k	0.2598	0.3562	0.2652	0.2937
v2-365k	0.2566	0.3664	0.2712	0.2981
v2-390k	0.2556	0.3438	0.2498	0.2831
v2-460k	0.2540	0.3778	0.2549	0.2956
v2-520k	0.2540	0.3778	0.2549	0.2956
v2-590k	0.2547	0.3651	0.2455	0.2884
v2-630k	0.2562	0.3632	0.2643	0.2946
v2-680k	0.2538	0.3740	0.2592	0.2957
v2-775k	0.2535	0.3750	0.2583	0.2956