Conversations in Galician: a Large Language Model for an Underrepresented Language

Eliseo Bao[1*], Anxo Pérez[1] and Javier Parapar[1]

[1]Information Retrieval Lab, Centro de Investigación en Tecnoloxías da Información e da Comunicación (CITIC), Universidade da Coruña, Campus de Elviña s/n, A Coruña, 15071, Galicia, Spain.

*Corresponding author(s). E-mail(s): eliseo.bao@udc.es;
Contributing authors: anxo.pvila@udc.es; javier.parapar@udc.es;

Abstract

The recent proliferation of Large Conversation Language Models has highlighted the economic significance of widespread access to this type of AI technologies in the current information age. Nevertheless, prevailing models have primarily been trained on corpora consisting of documents written in popular languages. The dearth of such cutting-edge tools for low-resource languages further exacerbates their underrepresentation in the current economic landscape, thereby impacting their native speakers. This paper introduces two novel resources designed to enhance Natural Language Processing (NLP) for the Galician language. We present a Galician adaptation of the Alpaca dataset, comprising 52,000 instructions and demonstrations. This dataset proves invaluable for enhancing language models by fine-tuning them to more accurately adhere to provided instructions. Additionally, as a demonstration of the dataset utility, we fine-tuned LLaMA-7B to comprehend and respond in Galician, a language not originally supported by the model, by following the Alpaca format. This work contributes to the research on multilingual models tailored for low-resource settings, a crucial endeavor in ensuring the inclusion of all linguistic communities in the development of Large Language Models. Another noteworthy aspect of this research is the exploration of how knowledge of a closely related language, in this case, Portuguese, can assist in generating coherent text when training resources are scarce. Both the Galician Alpaca dataset and Cabuxa-7B are publicly accessible on our Huggingface Hub, and we have made the source code available to facilitate replication of this experiment and encourage further advancements for underrepresented languages.

Keywords: large language model, conversational language model, low resource language, Galician, instructions

1 What is Cabuxa-7B?

Cabuxa-7B[1] is a LLaMA-7B [1] LoRA [2] instruct-tuned model for Galician that can answer instructions in the Alpaca format[2]. This work broadens the Portuguese effort from Larcher et al. [3] to Galician. Cabuxa-7B is intended to address a pressing need in the realm of low-resource languages, particularly for Galician. Low-resource languages often lack robust language models, making natural language processing tasks challenging in these linguistic contexts.

LLaMA, which stands for Large Language Model Meta AI, is a family of large language models introduced by Meta AI in February 2023. These models come in various sizes, including 7 billion, 13 billion, 33 billion, and 65 billion parameters.

Traditional fine-tuning of large language models for specific tasks can be prohibitively expensive in terms of computational resources. Low-Rank Adaptation of Large Language Models (LoRA) offers a novel approach to this problem. It involves preserving the pre-trained model's weights and introducing trainable rank decomposition matrices into each layer of the Transformer architecture [4]. This approach significantly reduces the number of trainable parameters for downstream tasks. In comparison to conventional fine-tuning, LoRA can reduce the number of trainable parameters by a factor of 10,000 and reduce the GPU memory requirements by threefold.

2 Data

We translated the Alpaca [5] dataset to Galician with the Python package `googletranslatepy`[3]. Cabuxa-7B was fed with the 80% of this new dataset[4], as we are keeping the remaining 20% for future evaluation and experiments.

3 Training procedure

We trained the model for 20 epochs with a Transformers [6] Trainer object. This object was configured with the following `TrainingArguments`:

- per_device_train_batch_size: 64
- gradient_accumulation_steps: 32
- warmup_steps: 100
- num_train_epochs: 20
- learning_rate: 3e-4
- fp16=True

LoRA and quantization [7] configurations are available both in the repository we release with this work[5] and the Huggingface model card. Table 1 shows the loss values for each training epoch.

[1] https://huggingface.co/irlab-udc/cabuxa-7b
[2] https://github.com/tloen/alpaca-lora/blob/main/templates/alpaca.json
[3] https://suqingdong.github.io/googletranslatepy/
[4] https://huggingface.co/datasets/irlab-udc/alpaca_data_galician
[5] https://gitlab.irlab.org/irlab/cabuxa

Table 1
Training loss for each epoch

Epoch	Loss
0.98	2.610
1.97	2.059
2.95	1.509
3.93	1.379
4.92	1.284
5.9	1.208
6.88	1.150
7.86	1.117
8.85	1.087
9.83	1.066
10.81	1.051
11.8	1.036
12.78	1.025
13.76	1.016
14.75	1.011
15.73	1.003
16.71	0.996
17.7	0.998
18.68	0.992
19.66	0.990

4 Future steps

Future work includes improving the translation of the dataset. It would also be desirable to be able to extend it to a wider variety of sources and tasks. Another important issue for the future is evaluation, which would ideally involve the use of expert linguists. Finally, we also intend to train and release versions of the model with a larger number of parameters.

5 Environmental Impact

The experiments were conducted using a private infrastructure. A cumulative of 72 hours of computation were performed on hardware of type NVIDIA RTX A6000. Total emissions are estimated to be 9.33 Kg. CO_2eq. Carbon emissions were estimated using the Machine Learning Impact calculator[6] presented by Lacoste et al. [8].

[6] https://mlco2.github.io/impact/

Appendix A How to get started with the model

Use the code below to get started with the model:

```
from peft import PeftModel
from transformers import AutoModelForCausalLM, LlamaTokenizer, GenerationConfig

config = PeftConfig.from_pretrained("irlab-udc/cabuxa-7b")
model = AutoModelForCausalLM.from_pretrained("huggyllama/llama-7b", device_map="
    auto")
model = PeftModel.from_pretrained(model, "irlab-udc/cabuxa-7b")
tokenizer = LlamaTokenizer.from_pretrained("huggyllama/llama-7b")

# This function builds the prompt in Alpaca format
def generate_prompt(instruction, input=None):
    if input:
        return f"""Abaixo está unha instrución que describe unha tarefa, xunto
    cunha entrada que proporciona máis contexto.
               Escribe unha resposta que responda adecuadamente a entrada.
               ### Instrución:
               {instruction}
               ### Entrada:
               {input}
               ### Resposta:"""
    else:
        return f"""Abaixo está unha instrución que describe unha tarefa.
               Escribe unha resposta que responda adecuadamente a entrada.
               ### Instrución:
               {instruction}
               ### Resposta:"""


def evaluate(instruction, input=None):
    prompt = generate_prompt(instruction, input)
    inputs = tokenizer(prompt, return_tensors="pt")
    input_ids = inputs["input_ids"].cuda()
    generation_output = model.generate(
        input_ids=input_ids,
        generation_config=GenerationConfig(do_sample=True),
        return_dict_in_generate=True,
        output_scores=True,
        max_new_tokens=256,
    )
    for s in generation_output.sequences:
        output = tokenizer.decode(s)
        print("Resposta:", output.split("### Resposta:")[1].strip())

evaluate("Cal é a fórmula química da auga?")
evaluate(
    "Convence ao lector por que é importante un determinado tema.",
    "Por que é esencial priorizar o sono?",
)
```

Listing 1 Cabuxa 7-B playground example.

References

[1] Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., Lample, G.: LLaMA: Open and Efficient Foundation Language Models (2023)

[2] Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: LoRA: Low-Rank Adaptation of Large Language Models (2021)

[3] Larcher, C., Piau, M., Finardi, P., Gengo, P., Esposito, P., Caridá, V.: Cabrita: closing the gap for foreign languages (2023)

[4] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention Is All You Need (2023)

[5] Taori, R., Gulrajani, I., Zhang, T., Dubois, Y., Li, X., Guestrin, C., Liang, P., Hashimoto, T.B.: Stanford Alpaca: An Instruction-following LLaMA model. GitHub (2023)

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

[7] Dettmers, T., Lewis, M., Shleifer, S., Zettlemoyer, L.: 8-bit optimizers via block-wise quantization. 9th International Conference on Learning Representations, ICLR (2022)

[8] Lacoste, A., Luccioni, A., Schmidt, V., Dandres, T.: Quantifying the Carbon Emissions of Machine Learning (2019)