Language	Retrieval Acc.
en	0.858
ar	0.492
bn	0.141
fi	0.756
id	0.680
sw	0.760
ko	0.453
te	0.056
ru	0.421

Table 4: Retrieval accuracy on TyDiQA dataset for chunk size = 100.

A.5 Handling Long Contexts

As discussed in §2.2, the models we study have limited context lengths and for QA tasks in particular, fitting the entire prompt containing the few-shot examples is often not feasible for low-resource languages where the tokenizers of these models are found to over-tokenize the text (nearly resulting in byte level tokens). To overcome this issue we follow two steps. First, for the few-shot examples we only provide the line within the paragraph containing the answer as the context. Second, for the test example, we index the chunks of the context using the embeddings from the `text-embedding-ada-002` model. Given the question, the closest chunk in the full context is retrieved and used in the prompt for the test example. We use a maximum chunk size of 100 in our experiments and use the implementation for retrieval provided in the **LangChain**[6] library. By doing this,we minimize the space taken by the context tokens in our prompt. Note that, for newer GPT models i.e. GPT-3.5-Turbo and GPT-4 which support longer context lengths, hence we only use this retrieval strategy for DV003 on QA tasks.

We attribute the significantly worse performance of DV003 on IndicQA to imperfect retrieval in the case of DV003, while for GPT-3.5-Turbo we do not rely on retrieval due to the larger context size. We provide the retrieval accuracies for DV003 (i.e. if the retrieved chunk contains the answer) in Appendix Table 4 , where we clearly see for low-resource languages like Telugu, the accuracies can be as low as 5%. While beyond the scope of this work, alternate retrieval strategies like using better embeddings from multilingual models for retrieval can be explored to close this gap (Nambi et al., 2023).

A.6 Factors Explaining Multilingual Capabilities of LLMs

We provide correlation plots in Figures 7 (between performance and fertility) and 8 (between performance and pre-training size) for both GPT-3.5-Turbo and GPT-4. The exact values of the correlations for all tasks and the two models is provided in Table 5.

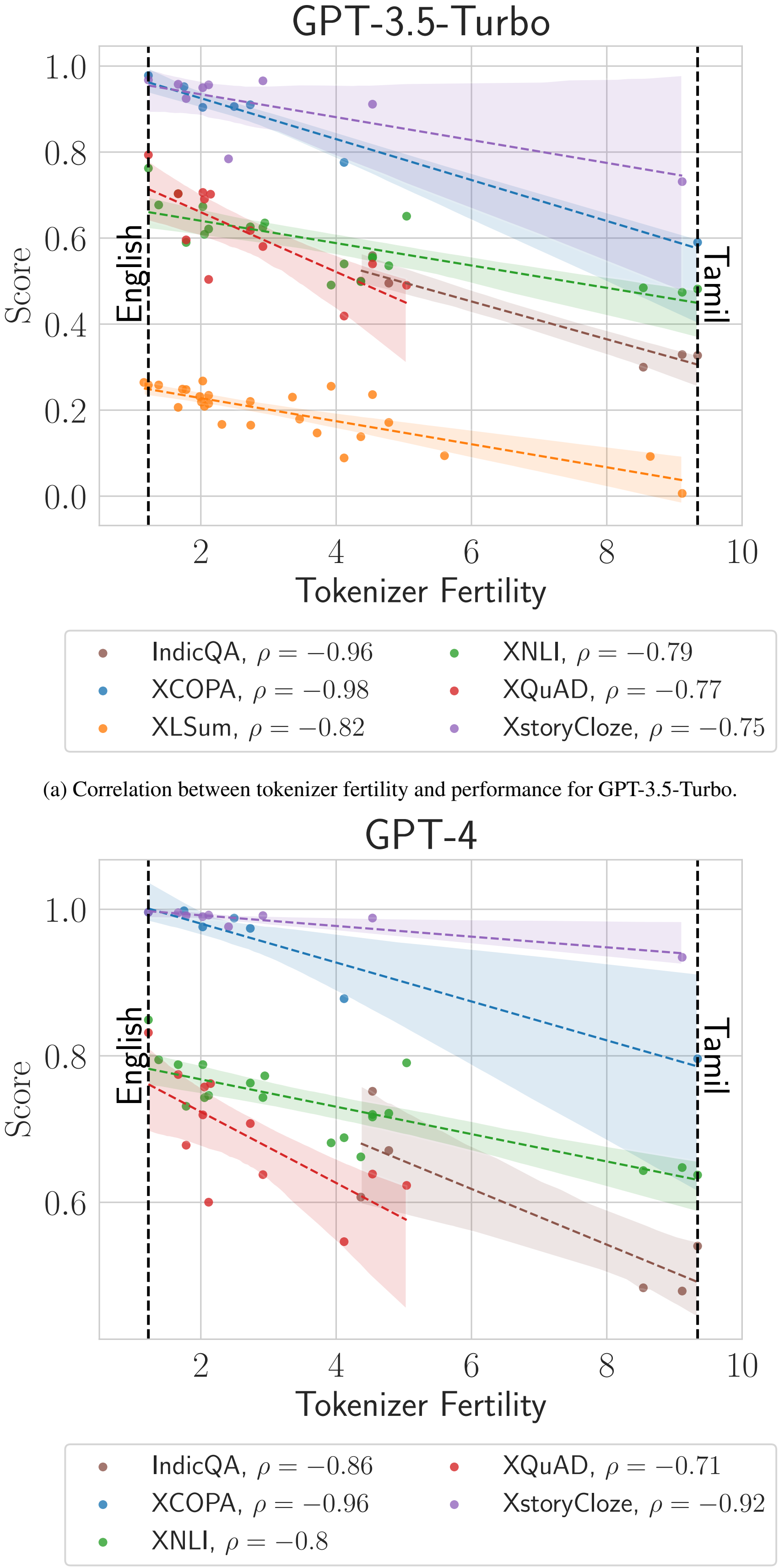
A.7 Challenges in Multilingual Evaluation

Effect of number of in-context examples k. Our main experiments were conducted with $k = 8$ or $k = 4$, depending on the task. Here, we evaluate what effect different numbers of in-context examples have on XNLI and XCOPA for three languages in Figures 6a and 6b. We observe while the performance increases sharply while moving from 0 to 2-4 examples, it is fairly stable after $k \geq 8$, with the exception of Haitian Creole in XCOPA, where it continues to improve.

Effect of language-specific prompt tuning. As discussed in §2.3.1, we use English validation data for prompt selection in each dataset that we use for all languages. Here, we explore whether separately tuning the prompts for each language helps. For XNLI, we run this experiment on Urdu and Swahili, tuning over ten different prompt templates from Prompt-Source, but find that the same prompt that was tuned for English gets picked up for these two languages as well. For XCOPA however, different prompts are chosen when tuned on Haitian Creole and Tamil. This leads to an improvement in the test performance for Haitian Creole (from 72% to 75.6%, see Figure 6c). Interestingly for Tamil, we see the test performance actually drops slightly compared to the accuracy obtained with prompt selected on English data, which we conjecture might be due to the fact that the validation sets in XCOPA have only 100 examples that may not be sufficient for selecting optimal prompts.

Effect of Explanations. Ye and Durrett (2022b), showed for `text-davinci-002`, that prompting the model with explanations before the outputs (Explain-then-Predict) in the in-context examples can help improve few-shot performance substantially on English language datasets. Hence, here we evaluate if they help improve the multilingual performance of the GPT-3.5-Turbo model as well. We perform experiments on XStoryCloze and XCOPA datasets and use the explanations available in Super-

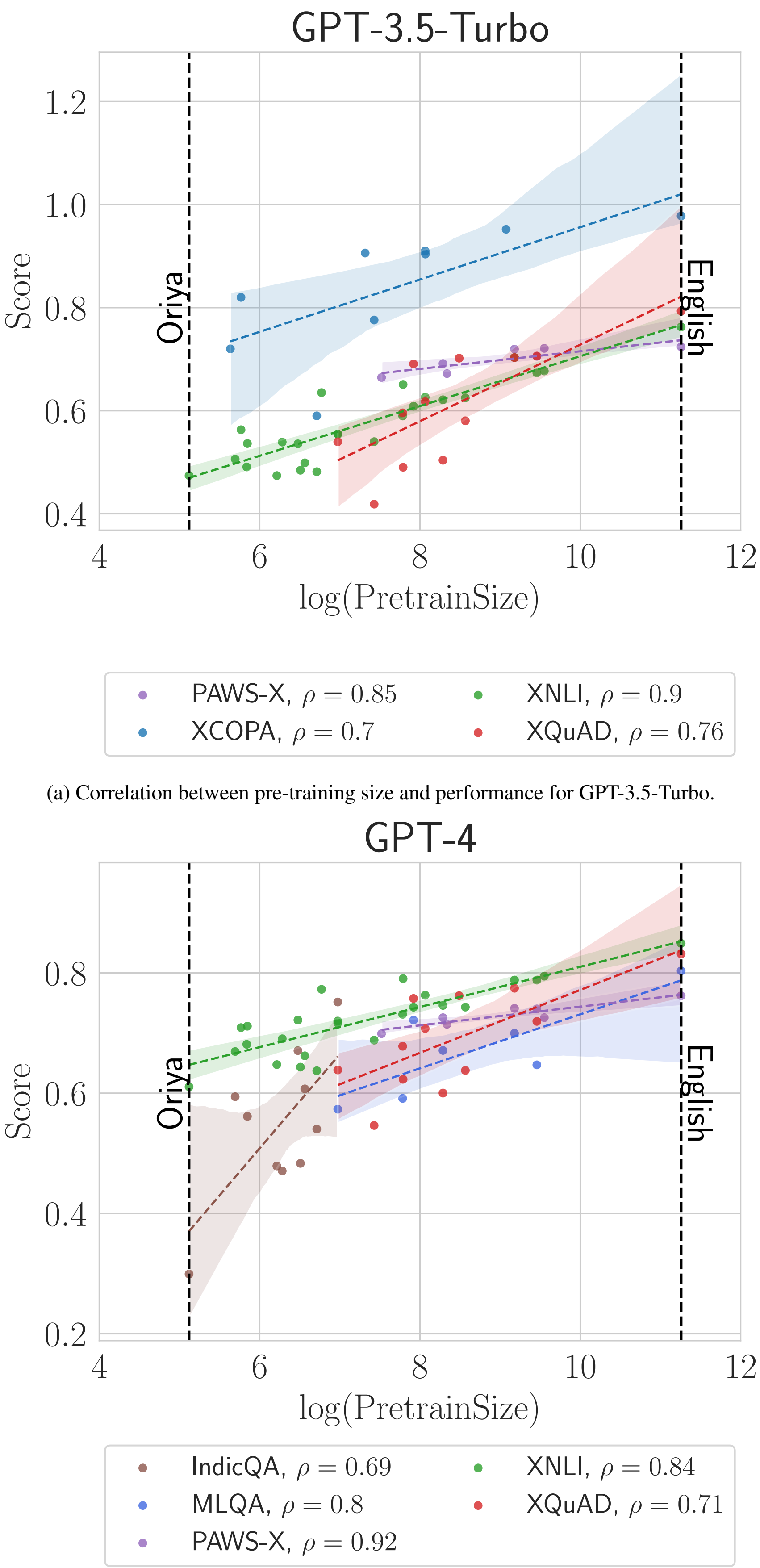
[6] https://github.com/hwchase17/langchain



(a) Correlation between tokenizer fertility and performance for GPT-3.5-Turbo.

(b) Correlation between tokenizer fertility and performance for GPT-4

Figure 7: Correlation between the performance of GPT-3.5-Turbo and GPT-4 with the tokenizer fertility.



(a) Correlation between pre-training size and performance for GPT-3.5-Turbo.

(b) Correlation between pre-training size and performance for GPT-4

Figure 8: Correlation between the performance of GPT-3.5-Turbo and GPT-4 with the pre-training size.