Enhancing Amharic-Llama: Integrating Task Specific and Generative Datasets

Israel Abebe Azime [1], **Mitiku Yohannes Fuge** [2], **Atnafu Lambebo Tonja** [3,4], **Tadesse Destaw Belay** [3], **Aman Kassahun Wassie**[2], **Eyasu Shiferaw Jada**, **Yonas Chanie**[5], **Walelign Tewabe Sewunetie**[6], **Seid Muhie Yimam** [7]

[∀] Masakhane NLP, [∀] Ethio NLP, [1] Saarland University, Germany, [2] AIMS, [3] Instituto Politécnico Nacional, Mexico, [4] Lelapa AI,[5] Carnegie Mellon University, [6] Debre Markos University, [7]Universität Hamburg, Germany

Abstract

Large language models (LLMs) have received a lot of attention in natural language processing (NLP) research because of their exceptional performance in understanding and generating human languages. However, low-resource languages are left behind due to the unavailability of resources. In this work, we focus on enhancing the LLaMA-2-Amharic model by integrating task-specific and generative datasets to improve language model performance for Amharic. We compile an Amharic instruction fine-tuning dataset and fine-tuned LLaMA-2-Amharic model. The fine-tuned model shows promising results in different NLP tasks. Our dataset creation pipeline , along with instruction datasets, trained models, and evaluation outputs, is made publicly available to encourage research in language-specific models [1].

1 Introduction

Large language models (LLMs) such as GPT series (Brown et al., 2020), Llama-2 (Touvron et al., 2023), Phi2 (Javaheripi et al., 2023), Mistral (Jiang et al., 2023), Mixtral (Jiang et al., 2024), PaLM (Chowdhery et al., 2023), Gemini (Team et al., 2023), BLOOM (Workshop et al., 2022), have exhibited exceptional performance in understanding and generating human language, showcasing a range of capabilities from basic linguistic comprehension to complex text generation.

Llama-2 (Touvron et al., 2023), a family of pre-trained and fine-tuned large language models (LLMs), demonstrated impressive performance across multiple tasks, particularly in dialogue-based interactions. Regardless of these achievements, Llama-2 pre-training supports a small number of languages, which does not include low-resource languages like Amharic. This makes adapting LLMs to low-resource languages that are not included a significant challenge.

Adopting these LLMs to local languages requires the preparation of a quality instruction dataset. Amharic is one of the Semitic languages under the Afroasiatic language family spoken in Ethiopia with more than 57M speakers. There are numerous task-specific datasets for Amharic (Tonja et al., 2023) compared to other Ethiopian languages. This paper focuses on enhancing the Llama-2-Amharic model[2] with quality datasets that are created by converting existing datasets in English into instruction-based Amharic datasets. Furthermore, we create new instruction datasets following the approach by Wei et al. (2021).

Llama-2-Amharic model was created by pre-training Llama-2 7B model using open-source Amharic and translated corpus. After performing vocabulary expansion and pre-training, the authors fine-tuned the created model by translating English instruction datasets into Amharic using commercial translation tools. In our research, we aim to improve the performance of the Amharic LLaMA model by integrating task-specific and generative datasets as shown in Table 1. The contributions of this paper are as follows:

[1] For data generation pipeline, see `https://github.com/EthioNLP/afri-sft-data`. For models and datasets, refer to `https://huggingface.co/EthioNLP`.

[2] `https://huggingface.co/iocuydi/llama-2-amharic-3784m`

- Creating Amharic instruction fine-tuning data from existing NLP task-specific and generative datasets.
- Evaluating new and existing models' performance.
- Exploring the effect of carefully curated datasets by combining them with machine-translated instruction datasets.
- Exploring the effect of instructions on the model's performance by introducing code-mixing instructions.
- Open-sourcing our dataset creation pipeline, instruction datasets, trained models, and evaluation outputs to promote language-specific studies on these models.

2 Related Work

The introduction of open-source LLMs like LLAMA-2 (Touvron et al., 2023) enabled the creation of several language models that focus on specific applications. This application gives more capabilities for these LLMs by teaching them to use tools (Schick et al., 2023), write code (Roziere et al., 2023), understand videos (Zhang et al., 2023a), or work for different languages (Cui et al., 2023). To achieve remarkable understanding and generation abilities, LLMs require large training data and huge compute resources (Hoffmann et al., 2022).

The work done by Dong et al. (2023) explores how LLMs' generation, natural language understanding, and problem-solving abilities relate to the data they are trained on and its composition. This work suggests that the amount of composition data is more important for these abilities to show in a low-resource scenario.

Training resources are one of the most important factors for adopting these models to specific languages. Techniques like low-rank adaptation (LoRA) (Hu et al., 2021) worked on reducing the number of trainable parameters using decomposition matrices during training. Quantized LoRA (QLoRA) (Dettmers et al., 2023) improves training time by introducing quantization and other numerical data type tricks. Using self-instructed fine-tuning, the work by Wei et al. (2021); Taori et al. (2023); Cui et al. (2023) showed a new approach to align the generation outputs of the generative models through the application of NLP tasks. These tasks are structured around natural language instruction templates, providing a novel means to guide the model's generation process toward better adherence to task-specific requirements. LLAMA-Adapter (Zhang et al., 2023b) also shows that it is possible to reduce the fine-tuning time of LLAMA-7B by introducing lightweight adapters on top of the model.

Acquiring and preparing a dataset for instruction fine-tuning presents a significant challenge due to the extensive labor and resources required. There are several ways of acquiring instruction data, including manual dataset curation, using generative models (Wang et al., 2022; Taori et al., 2023), or using machine translation instruction data for training LLMs for specific languages (Cui et al., 2023).

Fine-tuning LLMs such as LLAMA-2 for specific tasks is an area of exploration as well. Advanced language model-based translator (ALMA) (Xu et al., 2023) outperformed state-of-the-art (SOTA) no language left behind (NLLB) (NLLB Team et al., 2022) model MT task. They worked on fine-tuning monolingual data and subsequent fine-tuning with parallel data. Apart from LLAMA-2, Moslem et al. (2023) worked on Mistral 7B fine-tuning for medical domain machine translation, where they showed improvement in Spanish to English translation from baseline performance.

After the LLAMA-2 was released, researchers successfully adapted the model for other languages. The work by Cui et al. (2023) involved creating a unique tokenizer for Chinese, extending the pre-training phase, and then fine-tuning the model. This work incorporates secondary pre-training using Chinese data and fine-tunes the model with Chinese instruction datasets. The result shows a significant enhancement of the model's ability to comprehend and execute instructions.

To the same approach of the work by Cui et al. (2023), LLAMA-2 was also adopted for the Amharic (iocuydi, 2024) language. During pre-training, iocuydi (2024) used an open-source Amharic corpus with some translated corpus from English, and for fine-tuning, available English instruction datasets were translated to Amharic using the Google Translate API and SeamlessM4T. Following the increase of the LLAMA vocabulary size from 32k to 51k, and subsequent pre-training with a large

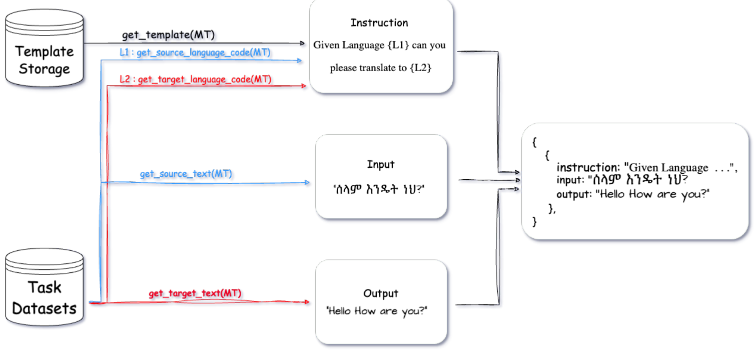
Template Storage
Task Datasets
get_template(MT)
L1 : get_source_language_code(MT)
L2 : get_target_language_code(MT)
get_source_text(MT)
get_target_text(MT)
Instruction
Given Language {L1} can you please translate to {L2}
Input
"ሰላም እንዴት ነህ?"
Output
"Hello How are you?"
{
{
instruction: "Given Language . . .",
input: "ሰላም እንዴት ነህ?
output: "Hello How are you?"
},
}

Figure 1: Data processing Pipeline.

Amharic text corpus, they conducted supervised instruction fine-tuning using machine-translated datasets. Then they evaluated their model using the MMLU (Hendrycks et al., 2020) multiple-choice English dataset by translating to Amharic. The model is available without original Amharic evaluations because no instruction-based datasets exist for Amharic.

Data Source	**Source Data**			**Is new**	**# Templates**	**Generated Data**		
	train	**val**	**test**			**train**	**val**	**test**
Amharic QA	1723	595	299	NO	14	10000	595	299
MasakhaNews	11522	188	376	NO	11	7866	205	376
MT (amh-eng)	497739	1012	1012	NO	10	10000	997	1012
MT (eng-amh)	497739	1012	1012	NO	10	10000	997	1012
Summarization	5761	719	719	NO	9	10000	719	719
Text Expansion	5761	719	719	NO	9	10000	719	719
Sentiment Analysis (AfriSenti)	5984	1497	1999	NO	7	10000	1728	1999
NER	1750	500	250	NO	9	10000	500	250
News Title Generation	-	-	-	Yes	10	10000	5078	5078
Poem Generation	-	-	-	Yes	3	3885	69	70
Poem Completion	-	-	-	Yes	7	3885	69	70
Religious Lyrics Generation	-	-	-	Yes	3	4929	188	206
Religious Lyrics Completion	-	-	-	Yes	4	10000	1497	1728
Story generation	-	-	-	Yes	10	1665	24	25
Spelling Correction	-	-	-	Yes (modified)	9	10000	1438	1438
Non religious music Lyrics Generation	-	-	-	Yes	4	148	5	5
Non religious music Lyrics Completion	-	-	-	Yes	7	259	5	5
Total						122,637	14,911	15,011

Table 1: **Dataset** used for preparing instruction fine-tuning data. **Is new** = new custom dataset.

3 Dataset preparation

In this work, we have converted existing NLP task-specific datasets, like sentiment analysis and machine translation, into instruction datasets. We created an instruction template for each task and developed a data creation pipeline that merges each template instruction with appropriate data from a pre-existing dataset. This approach helps us to create instruction datasets from pre-existing NLP task datasets. For the new NLP task, we focused on collecting a new dataset that can be converted into instruction data. We also created new datasets by tweaking existing datasets. Finally, we included an instruction-tuning dataset converted into Amharic language using machine translation systems. Table 1 shows a detailed distribution of instruction task data.

3.1 Instruction dataset from existing datasets

We have used several existing datasets to create an instruction dataset from an existing one. The production of these datasets includes web scraping, human labeling, and verification. By collecting and using this dataset for instruction, we ensure the quality of our instruction dataset. The other benefit of working with these datasets is that we ensure similar prompts across all our models for testing, which eliminates prompt-related performance variance that is usually reported while evaluating the performance of this dataset in generative LLMs.

For sentiment analysis data, we used *AfriSenti* (Muhammad et al., 2023), a sentiment analysis benchmark dataset for 14 African languages where Amharic is among the ones. The dataset is labeled with three sentiment classes: positive, negative, and neutral. The number of train, test, and val sets are shown in Table 1. We used the Amharic version of the classes for the test cases, and tests were done to check if the model gives one of the sentiment classes during generation.

We also worked with *MasakhaNews* (Adelani et al., 2023) which is a benchmark dataset for news topic classification covering 16 languages widely spoken in Africa. It provides an evaluation of baseline models from classical machine learning models and fine-tunes several language models.

To test if our model has the ability to identify names from sentences, we modified *MasakhaNER* (Adelani et al., 2021), which is a dataset for named entity recognition (NER) in ten African languages. For this work, we created questions to extract only personal names, and we plan to include more in our future works.

AmharicQA (Abedissa et al., 2023) is a publicly available Amharic open-ended question-answering dataset. It is a crowdsourced 2,628 question-answer pairs over 378 Wikipedia articles. These question-answer pairs are supplemented with context that the language model can use to answer the questions. We have also used this dataset to evaluate our models by converting it into an instruction dataset.

For tasks like Amharic text summarization, we used *XL-Sum* (Hasan et al., 2021), a comprehensive and diverse dataset comprising 1M annotated article-summary pairs. The dataset covers 44 languages, ranging from low to high-resource ones. We utilized the Amharic portion of the dataset in two ways. First, we took the text and prepared an instructional dataset to test our model's ability to summarize the text. We also created a text expansion task where our model takes the shorter sentence and produces a detailed explanation about the text, the inverse of the text summarization task.

Finally, to prepare training, validation, and testing for machine translation, we used the dataset in the works done by Barrault et al. (2019); NLLB Team et al. (2022). Our training dataset is from WMT19 (Barrault et al., 2019) and validation and testing are from NLLB Team et al. (2022).

The *Amharic spell correction* dataset is designed to assess the effectiveness of models in correcting Amharic spelling errors, covering common misspellings to advance NLP tools for the language. For this task, we leveraged Amharic BBC news texts from xlsum (Hasan et al., 2021). We also leveraged the text augmentation library nlpaug (Ma, 2019). We introduced some random character augmentations, including *insertion*, *substitution*, *swapping*, *deletion* and *word cropping*. This augmentation is done randomly and applied to part of the dataset.

After preparing each dataset, we found that the machine translation dataset we have was significantly larger than the other tasks, so we set maximum threshold of 10k instructions randomly for the training split of each dataset. For validation and testing, we only used one template per task, and we did not expand the data sizes.

3.2 New Custom Datasets

Most of the task datasets we prepared in Section 3.1 did not focus on generation tasks. Generation tasks are less explored for low-resource languages like Amharic, so we created original datasets collected from publicly available sources.

In Amharic, music, stories, and poems represent fascinating cultural artifacts. To facilitate the training and evaluation of models' capabilities in processing these tasks, we have created three new datasets. The first track we considered is *religious music lyrics generation*. There are several types of music lyrics we included in this dataset. We collected above 2k Amharic spiritual song lyrics from WikiMezmur [3]. Despite the popularity of non-religious music in Ethiopia, finding a freely available source to include this in our data was difficult; hence, our non-religious music data was

[3] https://wikimezmur.org/am

smaller than the others. To expand this dataset, we split the lyrics into verses and created a new completion task where the input is the first verse and the output is the remaining whole verse.

To understand the story generation abilities of different models, we created a dataset for *Ethiopian folktales*: We collected several Ethiopian folktales from EthopianFolkTales[4]. These stories are collected from all Ethiopian regions. Given that the dataset comprises traditional Ethiopian stories, there is no copyright restriction on them, and our usage is only for research purposes. We also collected *Amharic poem* from several public telegram channels.

For *news title generation*, we collected 50k news title and body pairs from different Amharic news sources such as BBC News[5], Deutsche Welle (DW) news[6], Sheger FM[7], Addis Admass Newspaper[8], and VOA Amharic[9]. To save GPT-4 credits, we did our testing only on the first 1300 items of this data.

3.3 Translated instruction fine-tuning dataset

During LLAMA model self-instructed fine-tuning (Touvron et al., 2023), instruction datasets like Alpaca (Taori et al., 2023) and dolly (Conover et al., 2023) have been widely used. In the work iocuydi (2024), machine translation systems were used to translate these datasets into Amharic instruction fine-tuning data. This method is used by most papers that try to adopt LLAMA models for their language, like Cui et al. (2023). For Amharic versions of alpaca and dolly datasets, we used datasets used by LLAMA-2-Amharic (iocuydi, 2024) training. We explored the effect of training a model by using only our relatively clean and human-verified data alone and in combination with this machine translation data.

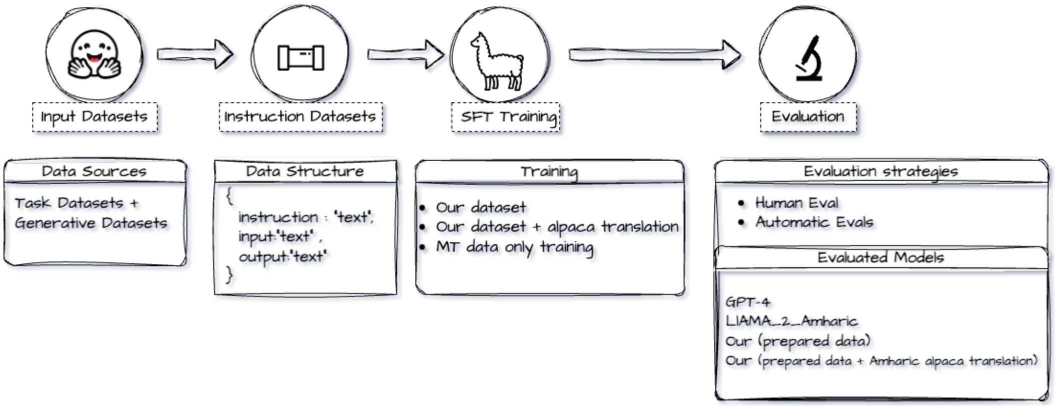

Figure 2: Full training Pipeline.

4 Experiments

We followed Chinese LLAMA (Cui et al., 2023) experiments to perform supervised fine-tuning (SFT) on our dataset using different types of the dataset we created. We used codes available on the Chinese-LLAMA-Alpaca[10] repository. We used 4, A100 GPUs with the default parameters in the repository. All training's are done for 3 epochs. All models and evaluation codes will be available in our repository. For MT task we also worked on M2M100 (Fan et al., 2021) and NLLB (NLLB Team et al., 2022) models.

[4] https://www.ethiopianfolktales.com/am
[5] https://www.bbc.com/amharic
[6] https://www.dw.com/am
[7] https://www.shegerfm.com/
[8] https://www.addisadmassnews.com/
[9] https://amharic.voanews.com
[10] https://github.com/ymcui/Chinese-LLaMA-Alpaca

During the evaluation of the models, we used `gpt-4-0613` for GPT-4. For our LLAMA-based models, we used a fixed generation parameters across the models.

Our main experiment includes:-

- Evaluating existing models on our dataset.
- Fine-tuning existing Amharic-LLAMA model using our own dataset.
- Exploring the effect of combining our dataset with existing machine-translated instruction datasets for Amharic.
- Fine-tuning existing models for only the machine translation dataset using our larger dataset and exploring.
- Exploring the effect of prompts in existing and available models for Amharic tasks.
- Exploring how code mixing affects the performance of models.

4.1 Datasets

The first set of experiments we conducted involved evaluating the base LLAMA-2-Amharic model (iocuydi, 2024) on our custom test set, which was created from different NLP task datasets. This will provide us with a baseline performance for Amharic tasks. The next set of experiments used different NLP task datasets that were converted into an instruction dataset by our data generation pipeline. We used LLAMA-2-Amharic model (iocuydi, 2024), which is pre-trained using the LLAMA-2 model for the Amharic language and performed supervised instruction fine-tuning on the task datasets. This ensures our model only has access to quality datasets that were adopted from verified NLP tasks. Finally, we combined our instruction dataset with the machine-translated instruction datasets. In the different datasets above, we have capped our training dataset to a maximum of 10k data from individual tasks, as shown in Table 1. We kept fixed instruction and data frequency in our validation and test set to avoid any performance variation because of instruction differences. For machine translation experiments, we created additional data that contains 200k data points from Barrault et al. (2019) and NLLB Team et al. (2022).

4.2 Evaluation Metrics

For selected NLP tasks in this paper, we used different evaluation metrics. For *sentiment analysis* and *news classification* tasks, we have used the weighted f1 score. For these classification tasks, we also keep track of the number of times the model returns output that cannot be classified as one of the classes.

For *generation tasks*, we used Rogue (Lin, 2004) scores. We used Rogue scores to evaluate *xlm-summarization*, *reverse summarization*, and *AmharicQA* tasks. We reported Rogue1, Rogue2, and RugueL metrics for generation tasks, but we heavily rely on RogueL for analysis since it focuses on the longest common subsequence rather than n-grams. We observed that most of our generation outputs do not share common n-grams when n is greater than 2, and the generations from systems like GPT-4 tend to be longer where the n-gram comparison methods express the results less. Additionally, we used Sacrebleu (Post, 2018) and chrf++ (Popović, 2017) automatic evaluation metrics for MT tasks.

Finally, we performed human evaluation for generative tasks such as music, poetry, and story generation. We sampled 120 individual items and did blind reviews using three people for each question. We created a rating system from 1 to 5 with detailed instructions, and we reported the average rating per task and model.

We did several evaluations for some tasks that were hard to evaluate, e.g., we used accuracy and Sacrebleu scores as evaluation metrics for AmharicQA following the suggestion by Abedissa et al. (2023); Lee et al. (2021). For tasks that require specific text output, we performed character normalization and text cleaning on the outputs before evaluation to avoid analysis because of typos and formatting issues.

In addition to the evaluation methods mentioned above, we explored the possibility of using GPT-4 for evaluation purposes, following the work from the Chinese LLAMA (Cui et al., 2023). Our assessment covered various generation tasks, showing that GPT-4 performs well in most areas. However, it shows inconsistency in scoring due to differences in the rating scale it assigns during each run. In addition, it struggles with evaluating poem and music generation tasks, as it does not fully understand Amharic poetic structure. Additionally, it encounters some challenges in evaluating machine translation, often missing grammatical details in Amharic sentences. Despite these limitations, GPT-4 has the potential for evaluating tasks if it is coupled with manual checks to ensure consistency. We expect similar difficulties in other low-resource languages based on our preliminary findings. While we did not include GPT-4 scores in our current reports due to time and cost constraints, we plan to include them in future research.

4.3 PROMPT BASED EXPERIMENTS

Throughout our investigation into sentiment analysis and news classification tasks in Amharic, we confronted instances where the model's outputs could not be classified into any of the defined categories. To overcome this issue, particularly in the GPT-4 model, we delved into the impact of customizing the prompts that guide the model. Our hypothesis centered around the idea that augmenting the Amharic prompt with concise English instructions could enhance the model's understanding, leading to more easily classifiable outputs.

In pursuit of this, we modified the prompt by introducing a succinct English description of the task before the Amharic prompt. This additional information clarified the expected format and content for the response. The additional English text stated, `"Below is an instruction that describes a task. Write a response that appropriately completes the request."`. Despite the seemingly modest nature of this adjustment, it yielded a significant reduction in unclassifiable outputs, which highlights the effectiveness of incorporating clear English instructions to steer the model toward the desired outcome.

5 RESULTS

Below, we discuss the performance of each model we tested by task type and evaluation strategy.

5.1 CLASSIFICATION RESULTS

For classification tasks, we used two metrics. Our models improve LLAMA-2-amharic scores as shown in Afrisenti, masakhanews and QA tasks in Figure 3. The other metrics we reported were how many times the model returns useful output. For AfriSenti classification task, 759 and 52 out of 1999 test data are not in any of the three classes for LLAMA-2-amharic and GPT-4, respectively. Our models reduce this unusable does not produce unusable outputs. For MasakhaNews 248, 136, 106, and 3, results are unusable for LLAMA-2-Amharic, our task dataset model, our combined model, and GPT-4, respectively. In MasakhaNews case, GPT-4 tends to take the lead in producing reasonable outputs.

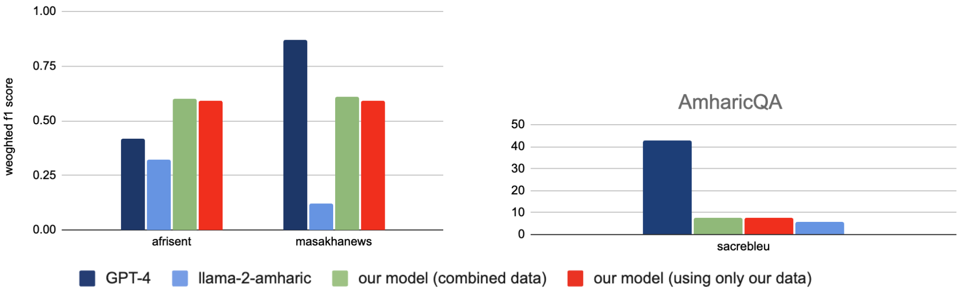

Figure 3: Generation scores: weighted f1 scores for afrisenti and masakhanews (left) and sacrebleu score for AmharicQA (right)

Tasks	GPT-4	LLaMA-2-amharic	ours (Task data)	ours (combined data)
Text summarization	3.41/0.11/3.34	0.61/0.00/0.62	1.12/0.00/1.13	0.78/0.00/0.80
Text expansion	3.11/0.11/3.10	3.35/0.02/3.22	2.14/0.02/2.05	2.89/0.10/2.82
Amharic QA	28.22/8.00/28.23	2.83/0.66/2.83	5.36/0.67/5.37	6.34/1.56/6.25

Table 2: **Rogue1/Rogue2/RogueL** scores for text summarization, Text expansion and AmharicQA

Tasks	GPT-4	LLaMA-2-amharic	ours (Task data)	ours (combined data)
Story generation	2.93	1.00	3.60	1.73
Poem completion	2.53	1.46	1.73	2.26
Poem generation	2.13	1.00	2.46	2.00
Religious Lyrics Generation	2.86	1.46	1.60	1.46
Religious Lyrics Completion	3.60	1.40	2.13	1.93
Non religious Lyrics Generation	3.53	1.00	1.60	2.06

Table 3: Average blind **human evaluation** out of 5, for three people in each task. **(1)** empty or non Amharic text. **(2)** not written in task format. **(3)** written in task format but no consistent idea and spelling errors. **(4)** looks like that specific generation task but has spelling and grammar errors. **(5)** this looks like a perfect generation of the task. Underlined text indicates cases where we see improvement compared to LLAMA-2-Amharic.

5.2 Generation Results

As explained in Section 4.2, we focus on RogueL metrics for our analysis. Across text summarization and AmharicQA, GPT-4 takes the lead, showing the generation ability of the model is very high. We were able to improve the LLAMA-2-Amharic model's ability for this task using our data, as shown in Table 2.

We conducted a human evaluation for the models that do not have fixed gold labels, as shown in Table 3. Table 3 result shows that the generation ability of LLLAMA-2-Amharic can be enhanced by adding generation-specific datasets. Our model lacks understanding of the specific formatting of texts because of the limitations in our pre-processing. However, it shows significant improvement where the LLAMA-2-amharic fails to understand the query.

5.3 Machine Translation (MT)

For the MT task, we evaluated two open-source sequence-to-sequence models: M2M100 (Fan et al., 2021) and NLLB (NLLB Team et al., 2022), GPT-4, LLAMA-2-amharic, and our models. Figure 4 shows sacrebleu and chrf++ results for the above MT models. As shown in the figure, from MT models, GPT-4 showed better results than the other when using English as the target language. However, our models showed results comparable to the NLLB and m2m100 models and outperformed the LLAMA-2-Amharic model for the Amharic-English translation direction. For the English-Amharic translation direction, NLLB model outperformed the others in the Sacrebleu score, while our models showed comparable results and outperformed GPT-4, LLAMA-2-Amharic and m2m100 models in this translation direction. In our MT evaluation, we noticed irregularities between the results of the two evaluation metrics. This shows that using only automatic evaluation metrics makes interpreting and generalizing the results hard. In the feature, we will add other metrics like human evaluation to evaluate MT results.

6 Conclusion and Future Works

In this work, we created Amharic instruction fine-tuning dataset, evaluated the performance of existing and our fine-tuned models in the new dataset, and explored the effect of carefully curated datasets on the models' performance. We observed a possibility of reusing task-specific datasets to improve the generation and task performance of the existing LLAMA-2-amharic model.

Our data generation pipeline that generates instruction datasets from task datasets can be used for the generation of similar datasets for other languages given template instructions. We are working on this kind of dataset for all languages included in MasakaNER (Adelani et al., 2021),

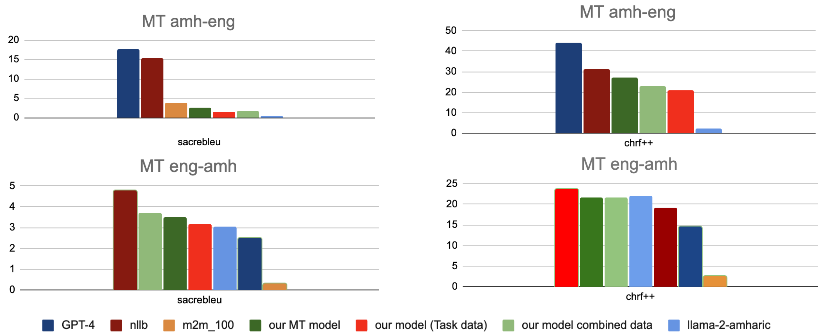

Figure 4: scores for **machine translation**. sacrebleu for Amharic to English translation (top left) chrf++ for Amharic to English translation (top right) sacrebleu for English to Amharic translation (bottom left) chrf++ for English to Amharic translation (bottom right)

MasakhaNews (Adelani et al., 2023), AfriSenti (Muhammad et al., 2023) and more to improve multilingual LLaMA models. We plan to open-source the instruction datasets with the generation code.

Moving forward, we aim to enhance both the quality and volume of the data utilized. Task-specific dataset creations are meant to complement, not replace, language-specific instruction dataset creations, and we plan to work on creating quality instruction datasets in addition to using existing task datasets. As discussed in section 4.2, we also plan to explore the relevance of and incorporate LLMs to evaluate our LLMs.

7 Limitations

One of the limitations we observed in our work is the lack of reliable generation metrics for our tasks. The models tend to generate wordy and explained outputs despite our attempts to specifically design the instruction template. As a solution, we used several metrics that can express one task's ability, and we reported the best-suited one.

In our current evaluation of all the models, we observed significant limitations while doing the spell correction and NER task. For Amharic spell correction all four generation models, including GPT-4, try to generate other things related to the text, and the word error rate for all of them is close to 99%.

We have yet to explore the effect of using machine-translated instruction datasets for building language-specific LLMs with regards to introducing cultural bias.

Acknowledgments

We are grateful to OpenAI for providing API credits through their Researcher Access API program to Masakhane for the evaluation of GPT-4 large language model.

References

Tilahun Abedissa, Ricardo Usbeck, and Yaregal Assabie. Amqa: Amharic question answering dataset. *arXiv preprint arXiv:2303.03290*, 2023.

David Ifeoluwa Adelani, Jade Abbott, Graham Neubig, Daniel D'souza, Julia Kreutzer, Constantine Lignos, Chester Palen-Michel, Happy Buzaaba, Shruti Rijhwani, Sebastian Ruder, et al. Masakhaner: Named entity recognition for african languages. *Transactions of the Association for Computational Linguistics*, 9:1116–1131, 2021.

David Ifeoluwa Adelani, Marek Masiak, Israel Abebe Azime, Jesujoba Oluwadara Alabi, Atnafu Lambebo Tonja, Christine Mwase, Odunayo Ogundepo, Bonaventure FP Dossou, Akintunde Oladipo, Doreen Nixdorf, et al. Masakhanews: News topic classification for african languages. *arXiv preprint arXiv:2304.09972*, 2023.

Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. Findings of the 2019 conference on machine translation (WMT19). In Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, André Martins, Christof Monz, Matteo Negri, Aurélie Névéol, Mariana Neves, Matt Post, Marco Turchi, and Karin Verspoor (eds.), *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pp. 1–61, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-5301. URL https://aclanthology.org/W19-5301.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240): 1–113, 2023.

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. Free dolly: Introducing the world's first truly open instruction-tuned llm, 2023. URL https://www.databricks.com/blog/2023/04/12/dolly-first-open-commercially-viable-instruction-tuned-llm.

Yiming Cui, Ziqing Yang, and Xin Yao. Efficient and effective text encoding for chinese llama and alpaca. *arXiv preprint arXiv:2304.08177*, 2023.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *arXiv preprint arXiv:2305.14314*, 2023.

Guanting Dong, Hongyi Yuan, Keming Lu, Chengpeng Li, Mingfeng Xue, Dayiheng Liu, Wei Wang, Zheng Yuan, Chang Zhou, and Jingren Zhou. How abilities in large language models are affected by supervised fine-tuning data composition. *arXiv preprint arXiv:2310.05492*, 2023.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, et al. Beyond english-centric multilingual machine translation. *Journal of Machine Learning Research*, 22(107):1–48, 2021.

Tahmid Hasan, Abhik Bhattacharjee, Md Saiful Islam, Kazi Samin, Yuan-Fang Li, Yong-Bin Kang, M Sohel Rahman, and Rifat Shahriyar. Xl-sum: Large-scale multilingual abstractive summarization for 44 languages. *arXiv preprint arXiv:2106.13822*, 2021.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

iocuydi. llama-2-amharic-3784m (revision 04fcac9), 2024. URL https://huggingface.co/iocuydi/llama-2-amharic-3784m.

Mojan Javaheripi, Sébastien Bubeck, Marah Abdin, Jyoti Aneja, Sebastien Bubeck, Caio César Teodoro Mendes, Weizhu Chen, Allie Del Giorno, Ronen Eldan, Sivakanth Gopi, et al. Phi-2: The surprising power of small language models, 2023.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

Hwanhee Lee, Seunghyun Yoon, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Joongbo Shin, and Kyomin Jung. KPQA: A metric for generative question answering using keyphrase weights. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2105–2115, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.170. URL https://aclanthology.org/2021.naacl-main.170.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-1013.

Edward Ma. Nlp augmentation. https://github.com/makcedward/nlpaug, 2019.

Yasmin Moslem, Rejwanul Haque, and Andy Way. Fine-tuning large language models for adaptive machine translation. *arXiv preprint arXiv:2312.12740*, 2023.

Shamsuddeen Hassan Muhammad, Idris Abdulmumin, Abinew Ali Ayele, Nedjma Ousidhoum, David Ifeoluwa Adelani, Seid Muhie Yimam, Ibrahim Sa'id Ahmad, Meriem Beloucif, Saif Mohammad, Sebastian Ruder, et al. Afrisenti: A twitter sentiment analysis benchmark for african languages. *arXiv preprint arXiv:2302.08956*, 2023.

Marta R. Costa-jussà NLLB Team, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. No language left behind: Scaling human-centered machine translation. 2022.

Maja Popović. chrf++: words helping character n-grams. In *Proceedings of the second conference on machine translation*, pp. 612–618, 2017.

Matt Post. A call for clarity in reporting bleu scores. *arXiv preprint arXiv:1804.08771*, 2018.

Baptiste Roziere, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, et al. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*, 2023.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *arXiv preprint arXiv:2302.04761*, 2023.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Atnafu Lambebo Tonja, Tadesse Destaw Belay, Israel Abebe Azime, Abinew Ali Ayele, Moges Ahmed Mehamed, Olga Kolesnikova, and Seid Muhie Yimam. Natural language processing in Ethiopian languages: Current state, challenges, and opportunities. In Rooweither Mabuya, Don Mthobela, Mmasibidi Setaka, and Menno Van Zaanen (eds.), *Proceedings of the Fourth workshop on Resources for African Indigenous Languages (RAIL 2023)*, pp. 126–139, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.rail-1.14. URL https://aclanthology.org/2023.rail-1.14.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language model with self generated instructions. *arXiv preprint arXiv:2212.10560*, 2022.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

BigScience Workshop, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, et al. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, 2022.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. A paradigm shift in machine translation: Boosting translation performance of large language models. *arXiv preprint arXiv:2309.11674*, 2023.

Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023a. URL https://arxiv.org/abs/2306.02858.

Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*, 2023b.