AfroCS-xs: Creating a Compact, High-Quality, Human-Validated Code-Switched Dataset for African Languages

Kayode Olaleye[*α] **Arturo Oncevay**[*β] **Mathieu Sibue**[*β]
Nombuyiselo Zondi[α] **Michelle Terblanche**[α] **Sibongile Mapikitla**[α] **Richard Lastrucci**[α]
Charese H. Smiley[β] **Vukosi Marivate**[α,γ]

[α]Data Science for Social Impact, University of Pretoria [γ]Lelapa AI
[β]JPMorgan AI Research
{kayode.olaleye,vukosi.marivate}@up.ac.za

Abstract

Code-switching is prevalent in multilingual communities but lacks adequate high-quality data for model development, especially for African languages. To address this, we present AfroCS-xs, a small human-validated synthetic code-switched dataset for four African languages (Afrikaans, Sesotho, Yoruba, isiZulu) and English within a specific domain—agriculture. Using large language models (LLMs), we generate code-switched sentences, including English translations, that are rigorously validated and corrected by native speakers. As a downstream evaluation task, we use this dataset to fine-tune different post-trained LLMs for code-switched translation and compare their performance against machine translation (MT) models. Our results demonstrate that LLMs consistently improve in translation accuracy when fine-tuned on the high-quality AfroCS-xs dataset, highlighting that substantial gains can still be made with a low volume of data. We also observe improvements on natural code-switched and out-of-domain (personal finance) test sets. Overall, regardless of data size and prior exposure to a language, LLMs benefit from higher quality training data when translating code-switched texts in under-represented languages.

1 Introduction

In today's digital era, the importance of multilingual capabilities in global communication, and more so in computational linguistics, is paramount. The phenomenon of code-switching[1]—where languages are alternated within a single discourse (Poplack, 1980)—presents a specific challenge in multilingual societies. It is highly present in African languages (Amuzu and Singler, 2014), which, despite their vast linguistic diversity, are often overlooked in computational linguistic studies (Nekoto et al., 2020; Marivate et al., 2020; Orife et al., 2020). Tailoring language technologies, such as MT systems, to accommodate these languages can enhance community empowerment and facilitate better communication, thereby advancing global digital inclusiveness.

This research aims to bridge this gap by proposing a methodology to create few but high-quality Afro-centric code-switched texts between English and multiple African languages, including Yoruba (yo), isiZulu (zu), Sesotho (st), and Afrikaans (af), to support language technologies development. These languages were selected to represent diverse linguistic families, geographical distributions, sociolinguistic contexts, and levels of exposure in NLP research.[2] To evaluate the effectiveness of this methodology, we compare the adaptation capabilities of LLMs with MT downstream tasks using a small curated set of $\sim$120 synthetic code-switched sentences for each African language, along with their English translation. These code-switched examples are initially generated using the in-context learning (ICL) capabilities of GPT-4 and *subsequently refined through human validation*. For a detailed examination of the dataset created, refer to Section 3. With the validated data, we benchmark various adaptations of BLOOMZ (Muennighoff et al., 2022), Llama-3 (Dubey et al., 2024), Aya-23 (Aryabumi et al., 2024)—which are multilingual instruction-tuned LLMs—and NLLB (Team et al., 2022) along with MADLAD-400 (Kudugunta et al., 2023)—which are massive translation-only models for 202 and 419 languages respectively.

Our research also compares the downstream performance of models trained on human-validated synthetic code-switched data against that of models trained on a larger mix of both human-validated

[*]These authors contributed equally to this work.
[1]In this paper, "code-mixing" and "code-switching" are used interchangeably to refer to the same concept.

[2]Additionally, at least one of the co-authors is a native or proficient speaker of each language, enabling rigorous validation and ensuring the authenticity of the dataset.

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 33391–33410

and non-validated synthetic data. Drawing on Xu and Yvon (2021), which highlights the benefits of machine-generated training data for French–English and Spanish–English code-switching in MT systems, we explore these findings for under-represented African languages.

Our contributions are as follows: (i) we underscore how our methodological approach and focus on African languages contribute to and diverge from existing literature; (ii) we introduce a framework for generating synthetic code-switched datasets, tailored specifically for African languages; (iii) we release new human-validated synthetic code-switched datasets in four African languages, totaling 479 human-validated and 3160 non-validated synthetic instances; and (iv) we analyse how LLMs benefit from small human-validated and large non-validated synthetic code-switched data for translation tasks.

For this study, we mainly focus on the agricultural domain, which is relevant to the speaker communities; however, our code-switched data generation approach is adaptable across topics. By contributing agriculture-related data, we aim to improve information access by converting agricultural knowledge and market data into code-switched format, making key concepts clearer for farmers in multilingual areas and aiding language learners to transition between their native language and the target language (van Gompel et al., 2014). To gauge the robustness to topic shift of models trained on the agriculture-focused data of AfroCS-xs, we also curate an out-of-domain, natural code-switched test set tackling personal finance matters in Yoruba.

2 Related Work

Mitigating the scarcity of code-switched datasets There is a general scarcity of code-switched data, with most existing datasets involving non-African languages. Consequently, most studies in machine translation of code-switched texts have largely overlooked African languages (Appicharla et al., 2021; Gupta et al., 2021; Nagoudi et al., 2021), leaving a significant gap that underscores the need for more research focused on African languages.

To address this data scarcity, several researchers have turned to synthetic data generation as a solution. For example, Song et al. (2019) and Xu and Yvon (2021) generate artificial code-switched data to enhance machine translation systems. This approach is particularly relevant to our study, which also creates synthetic data to tackle the lack of translated code-switched data in African languages. In particular, our work builds on recent studies that examine the potential and limitations of in-context learning in code-switched data generation and modelling (Lee et al., 2022; Rubin et al., 2022; Liu et al., 2022; Raunak et al., 2023; Yong et al., 2023; Wan et al., 2023; Terblanche et al., 2024). We employ GPT-4 to generate code-switched data suitable for fine-tuning and testing smaller LLMs for MT tasks.

Code-switching MT methodology and models used The methodologies employed in previous studies vary, with a common reliance on sequence-to-sequence learning and models like mT5, mBART, and gated convolutional models. For example, Jawahar et al. (2021) and Gautam et al. (2021) use mT5 and mBART for translating English to Hinglish, indicating a trend toward leveraging advanced neural models for code-switched translation tasks. Despite these advancements, the application of ICL and parameter-efficient fine-tuning remains less explored.

3 Synthesising AfroCS-xs

3.1 LLM-assisted data generation approach

Our data generation pipeline capitalises on the ICL (Dong et al., 2022) capability of GPT-4, necessitating the inclusion of five "shots" or examplars in the prompts.

Generating examplars The process begins by synthesising five initial examplars of code-switched sentences per language. For this, we provide GPT-4 with a detailed description of our objectives, effectively instructing the system on the specific types of sentences we aim to produce (Prompt A, Table 4). The resulting code-switched sentences are then reviewed by a native bilingual speaker for each of the four language pairs to guarantee both linguistic precision and authenticity.

Prompt design Our approach to generating code-switched data with GPT-4 relies on ICL using the five initial exemplars of code-switched sentences per language. We designed the prompt to frame GPT-4 as a game developer creating an educational farming game for youth. The prompt begins with: "*You are a game developer who wants to build an interactive and educational game about Agriculture and farming for youth...*" (See Prompt B, Table 4, in Appendix A for full prompt). This framing

Synthetic Yoruba–English (Agriculture)	
yo–en (m): Lọwọlọwọ, o nilo lati ṣe focus lori bi a ṣe le ṣe improve awọn ọna ogbin rẹ.	
yo–en (hm): Lọwọlọwọ, o need lati focus lori bi o ṣe le improve farming methods rẹ.	
translation: Currently, you need to focus on how to improve your farming methods.	
Natural Yoruba–English (Personal Finance)	
yo–en (ood): O wa important lati diversify loan rẹ lati minimize risk ti o wa ninu peer-to-peer lending.	
translation: It's important to diversify your loans to minimise risk in peer-to-peer lending.	

Table 1: Examples of synthetically generated, human-validated, and natural Yoruba–English code-switched sentences with English translations. **yo–en (m)** represents the machine-generated (synthetic) Yoruba-English sentence, while **yo–en (hm)** represents the human-validated version, where words in red were removed or replaced, and words in green indicate the replacements or insertions. **yo–en (ood)** refers to a natural Yoruba–English sentence in the personal finance domain, which is out-of-domain (OOD) relative to the agriculture-focused training data.

reflects a forward-looking motivation: to encourage more relatable and engaging sentence generation, especially for a future application aimed at agricultural education. Code-switching can make technical content more accessible to older, less formally educated farmers, while also resonating with younger bilingual users who are often disengaged from agriculture. While not central to our experiments, this framing helped steer GPT-4 toward more natural and context-aware outputs.

To enhance the utility of the generated code-switched sentences for subsequent supervised learning, we generate English translations for each synthetic code-switched sentence using GPT-4 (Prompt C, Table 4 in Appendix A) and validate them with native speakers. By providing these translations, the methodology effectively creates a bilingual corpus, which can then be used for fine-tuning other models in a low-resource setting to better understand and generate code-switched language content.

Human correction and validation We incorporate a human-in-the-loop approach to ensure the accuracy and cultural alignment of the code-switched sentences generated by GPT-4 and their English translations. We engage native speakers for correction and validation. Annotators are presented with both the code-switched sentences and their English counterparts, and are tasked with aligning them accurately using the guidelines in Appendix B. They focus on adjusting the sentences to reflect authentic code-mixing patterns and relevant cultural contexts. This process is crucial in refining the AfroCS-xs dataset, ensuring that the translations not only match linguistically but also resonate with the cultural nuances inherent to the native languages. An example of an LLM-generated Yoruba–English[3] sentence before and after human validation can be found in Table 1, block 1.

3.2 Natural and out-of-domain code-switch test sets for Sesotho and Yoruba

We curate two additional code-switch datasets in Sesotho–English and Yoruba–English for different testing purposes. For Sesotho–English, we provided a native speaker annotator fluent in both languages with 250 synthetically-generated English sentences covering agricultural topics. The annotator was instructed to rewrite each sentence into natural Sesotho–English code-switching, prioritising linguistically natural blends of both languages. The Yoruba–English dataset follows the same creation protocol for 334 instances but focuses on personal finance topics to enable an out-of-domain evaluation of the models. Both annotators were explicitly directed to avoid atypical mixing and mimic real-world code-switching usage. See Table 1, block 2 for an example of a natural, OOD Yoruba–English sentence, and Appendix E for more details about the dataset creation.

3.3 AfroCS-xs statistics

Overview Table 2 summarises code-switching metrics for the language pairs in AfroCS-xs.[4] The M-index (Guzmán et al., 2017; Barnett et al., 2000) quantifies the ratio of languages in the corpora; values closer to 1 indicate an equal distribution of

[3] For simplicity, only the accent below the alphabets is used in the Yoruba text, instead of the full set of diacritics.

[4] We also computed SyMCoM (Kodali et al., 2022) to assess syntactic complexity, but its reliability is limited by noisy POS tags for tokens, as produced by spaCy (Honnibal et al., 2020). These outputs would require manual correction and validation before they can be reliably used to compute SyMCoM scores; we therefore leave this for future work.

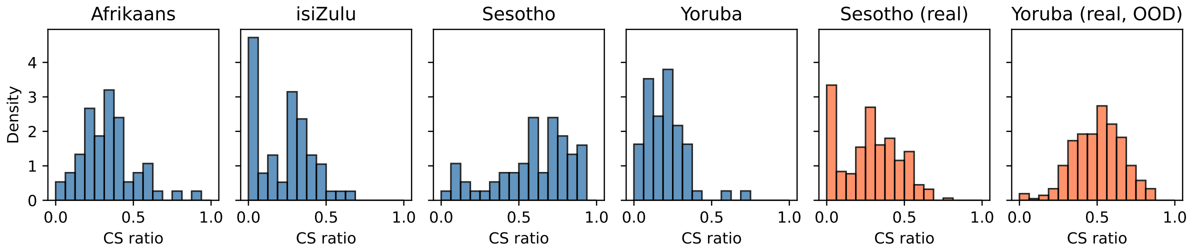

Figure 1: Distribution of English word ratios in code-switched sentences across AfroCS-xs (test sets) and real code-switched and out-of-domain data. Y-axis shows probability density, where the area under each curve sums to 1.

	M-Index	I-Index	Burstiness
af–en (m)	0.9999635	0.5021021	-0.1797550
af–en (hm)	0.9953195	0.4823695	-0.1591807
zu–en (m)	0.9045894	0.1534247	-0.2804299
zu–en (hm)	0.8979139	0.1542662	-0.2924524
yo–en (m)	0.5386369	0.3178893	-0.0470404
yo–en (hm)	0.6416407	0.3539823	-0.0532718
yo–en (ood)	0.9878679	0.4439600	-0.1563924
st–en (m)	0.9994678	0.2169982	-0.1321809
st–en (hm)	0.9982484	0.2203492	-0.1362026
st–en (real)	0.6246004	0.2497561	-0.0897291

Table 2: Code-switching metrics across language pairs (*m*: machine-generated, *hm*: human-validated, *real*: naturally occurring, *ood*: real and out-of-domain).

tokens across the languages. The I-index (Gambäck, 2014; Guzman et al., 2016; Gambäck and Das, 2016) measures the average number of switch points in a language, or the probability any given token is a switch point. Burstiness (Goh and Barabási, 2008) measures whether code mixing occurs in bursts or periodically. Negative burstiness values indicate predictable switching patterns, while positive values indicate unpredictable switching.

We observe that af–en and st–en pairs have the highest M-index, indicating a balanced use of both languages, while yo–en has the lowest. The af–en pairs also have the highest I-index, suggesting more frequent code-switching, whereas zu–en has the lowest, indicating less switching. All pairs show negative burstiness values, reflecting predictable switching patterns.

The natural Yoruba–English data (yo–en (ood)) has a higher M-index than its synthetic counterparts, indicating a more balanced mix of languages in financial discourse. Its I-index is also slightly higher, showing frequent switching, while its burstiness remains closer to that of synthetic data. The natural Sesotho–English data (st–en (real)) have a lower M-index than its synthetic versions, suggesting that one language is more dominant in real-world use but with a comparable I-index.

The code-switch ratio distributions in Figure 1 represent the distribution of English word ratios in code-switched sentences and further supports the M-index patterns previously observed. The natural Sesotho–English dataset skews left, reinforcing its lower M-index and the dominance of one language in real-world use. In contrast, the natural, out-of-domain Yoruba–English dataset (yo–en (ood)) is more evenly distributed, possibly due to greater domain-specific borrowing of terminology.

Part-of-speech (POS) distribution Table 3 presents the POS ratios for the code-switched English segments in the AfroCS-xs and complementary test sets.[5] As expected, nouns are the most frequent POS across all languages. However, a notable feature is the higher proportion of adjectives compared to verbs in the natural code-switched test sets (Sesotho and Yoruba) compared to their AfroCS-xs counterparts. This discrepancy may be attributed to the GPT-4's tendency to use code-switched English for actions (verbs), which annotators retained, even though in natural code-switching, speakers may prefer not to use English for actions.

POS	af	zu	st	yo	st-real	yo-ood
Noun	56.5	38.8	37.8	62.1	52.6	51.0
Verb	19.2	16.3	21.2	20.7	6.9	14.6
Adjective	14.0	10.7	11.6	9.7	20.1	20.7
Preposition	4.8	20.2	14.6	4.1	8.6	4.5
Proper Noun	1.8	0.6	4.8	2.1	7.4	6.0
Adverb	3.7	2.2	4.5	1.4	2.4	2.5
Conjunction	0.0	8.4	0.0	0.0	1.5	0.3
Pronoun	0.0	2.8	5.5	0.0	0.5	0.3

Table 3: Part of Speech distribution (%) of code-switched English segments across AfroCS-xs (test sets) and natural code-switched test sets.

[5]We use the Greedy Averaged Perceptron tagger, as implemented by Matthew Honnibal (https://explosion.ai/blog/part-of-speech-pos-tagger-in-python) and group its categories into the eight listed POS classes.

Summary Overall, the AfroCS-xs synthetic data aligns well with natural code-mixed corpora in key areas, including POS distributions and language-switching frequencies (I-index, burstiness). However, some discrepancies remain—such as imbalanced language ratios in Sesotho–English and an overuse of English verbs—highlighting that, while LLMs like GPT-4 can capture general code-switching structures, they may misrepresent finer linguistic patterns. These findings suggest room for improving prompting strategies to enhance the fidelity of synthetic code-mixed data, especially for low-resource language pairs.

4 Experimental Setup

4.1 Datasets

Human-validated data We perform experiments on the human-validated synthetic data that we outlined in Section 3. A total of 119, 120, 120 and 120 samples of parallel English sentences and Yoruba–English, isiZulu–English, Sesotho–English and Afrikaans–English code-mixed sentences were collected, respectively. From these, we split half (60 each) for low-resource fine-tuning, and use the rest for testing. We consider two main downstream tasks for evaluation: (i) translating English text to code-switched text (*EN2CS*) and (ii) translating code-switched text to English text (*CS2EN*). For CS2EN, the training set consists of pairs of code-switched texts and their corresponding English translations. For the model training process, structured input prompts are constructed, with each input prompt being derived from a template (Table 6, Appendix D) where the code-switched text and its English translation are populated. For EN2CS, we adjust the prompt template in Table 6 to reflect translating English text to code-switched text.

Extra synthetic data for training We also generate extra synthetic data, following the described methodology, but *without subsequent human validation*. The purpose of this additional data is to analyse whether LLMs and/or MT models can improve their downstream performance with extra data non-refined by human experts. The size of the extra synthetic data is 790 sentences per language pair.

4.2 LLM experiments with BLOOMZ, Llama-3, and Aya-23

BLOOMZ (Muennighoff et al., 2022), Llama-3 (Dubey et al., 2024), and Aya-23 (Aryabumi et al., 2024) are multilingual LLMs. The extensive post-training they have undergone enhances their ability to generalise to new tasks in a zero-shot setting, making them suitable for translating code-switched sentences in under-represented languages. For the BLOOMZ models, we note that only Sesotho, isiZulu and Yoruba were included in the pre-training data. All four languages are absent from the Aya-23 training data. As for Llama-3, since few details about its training data have been released, determining whether any of the four languages in scope were included is challenging.

Few-shot setup details We include few-shot (FS) experiments with 5 and 10 examples randomly sampled, averaging results over 3 seeds.

Fine-tuning BLOOMZ, Llama-3, and Aya-23 with QLoRA We leverage parameter-efficient fine-tuning through Low-Rank Adaptation (LoRA) (Hu et al., 2021), which introduces trainable low-rank matrices to specific modules. LoRA allows targeted adjustments to pre-trained models without retraining the entire model, significantly reducing computational requirements. We fine-tune quantized versions of BLOOMZ-3B, Llama-3-8B-Instruct, and Aya-23-8B with LoRA for the code-mix MT tasks.

Fine-tuning setup details We follow the procedure described in Section 4.1 and Table 6 (Appendix D) to transform the input sentence pairs into single prompts that we then tokenise and use for fine-tuning our models. We configure the learning rate to 1e-3 and employ the Adam optimizer. A batch size of 24 is used, and the evaluation is performed every 1000 steps. We halve the training set for validation in order to select a stopping point for the fine-tuning. Once we identify the best stopping point using the validation split, we re-train until this point using the full data. For the LoRA configuration, the rank for the low-rank approximation is set to 16, and the scaling factor for the low-rank adaptation is set to 32. The trainable parameters are limited to the self-attention layers of the model. Additionally, a dropout rate of 0.05 is applied in the LoRA layer. The model weights are quantized to 8-bit precision for BLOOMZ-3B and 4-bit precision for Llama-3-8B-Instruct and Aya-23-8B to reduce memory requirements. Mixed-precision training is enabled, using a combination of float16 and float32 data types to accelerate the training process. The fine-tuning code is implemented using the Hugging

Face Transformers library (Wolf et al., 2020) with NVIDIA Titan RTX & A10 24GB GPUs, aligning with the memory and computational requirements of the large models.

4.3 MT baselines

NLLB is a suite of large multilingual translation models for 202 languages (Team et al., 2022) and MADLAD-400 a suite trained with data spanning 419 languages (Kudugunta et al., 2023). Their pre-training data includes the four targeted languages in this experiment, making both models robust baselines for the translation tasks. We use the NLLB-200-3.3B and MADLAD400-3B-MT variants in zero-shot.

Fine-tuning considerations for MT systems We do not include fine-tuning results for NLLB-200-3.3B and MADLAD-400-3B-MT, as the scores we obtained in this setting were not statistically different from zero-shot ones (±1 point).[6] A possible explanation is that MADLAD-400-3B-MT and NLLB-200-3.3B were already trained on extensive multilingual data that included the four African languages in the scope of this work. It might be that subsequently fine-tuning them on a smaller-scale, domain-specific sample like AfroCS-xs offers limited benefits given the broader understanding of Yoruba, IsiZulu, Afrikaans, Sesotho, and English these models may have acquired during their prior training.

4.4 Evaluation metrics and copy baseline

As in previous code-switch translation work, we assess performance using the chrF++ and BLEU[7] metrics introduced in sacreBLEU (Post, 2018). This is followed by additional quantitative and qualitative analyses of the results and outputs. Given the context of code-switch experiments, we also include a **copy baseline** score, where the input sentence is used as the translation output. This helps measure the extent to which models modify parts of the text that they were not expected to alter.

It is worth noting that embedding-based metrics, such as COMET (Rei et al., 2020), correlate better with human judgment than chrF++ and BLEU (Kocmi et al., 2021). However, COMET models, and their more focused African variant AfriCOMET (Wang et al., 2024), only support Afrikaans and Yoruba—not Sesotho and isiZulu. As a complementary perspective, we thus provide partial COMET results in Figures 6 and 7 of Appendix F.

5 Results and Discussion

5.1 English to code-switched text (EN2CS)

Figure 2 illustrates the performance of translating English into code-switched text. In the baseline evaluation, MT models generally exhibit strong performance, particularly excelling in most language pairs. However, they are outperformed in three out of four languages by fine-tuned LLMs, highlighting the potential of LLMs when enhanced with high-quality data. Notably, the copy baseline is surpassed by LLMs in almost all cases, indicating that these models can effectively generate meaningful translations rather than copying the source text.

Moreover, the results seem to indicate fine-tuning (FT) is more effective than few-shot (FS) learning for improving LLM performance in code-switched translation tasks. Fine-tuning with human-validated data only (FT=H) consistently enhances the performance of LLMs, with BLOOMZ-3B showing remarkable improvements, especially in the Yoruba language. Aya-23-8B also demonstrates significant gains, benefiting more from the validated synthetic data than other LLMs, although this advantage is less pronounced for Afrikaans.

When incorporating additional *non-validated* synthetic data (FT=H+S), Aya-23-8B stands out as the only LLM to consistently benefit, particularly in Sesotho and Yoruba—suggesting its adaptability to synthetic enhancements. In contrast, BLOOMZ-3B and Llama-3-8B-Instruct do not exhibit the same level of improvement, and in some cases suffer a performance loss, underscoring the key role of data quality.

Overall, these results emphasize the importance of high-quality, human-validated data in fine-tuning LLMs for code-switched translation tasks. While MT models provide a strong baseline, LLMs demonstrate a robust capacity to learn and improve with targeted fine-tuning, although the utility of synthetic data varies across models and languages.

5.2 Code-switched text to English (CS2EN)

Figure 3 presents the results when translating from code-switched text into English. Given that part of the source sentence is already in English, and con-

[6] This contrasts with the significant gains we observe when fine-tuning LLMs with very small data in Section 5.

[7] We report chrF++ scores in the main paper, as they correlate more strongly with human judgment (Kocmi et al., 2021), with BLEU results provided in Appendix F.

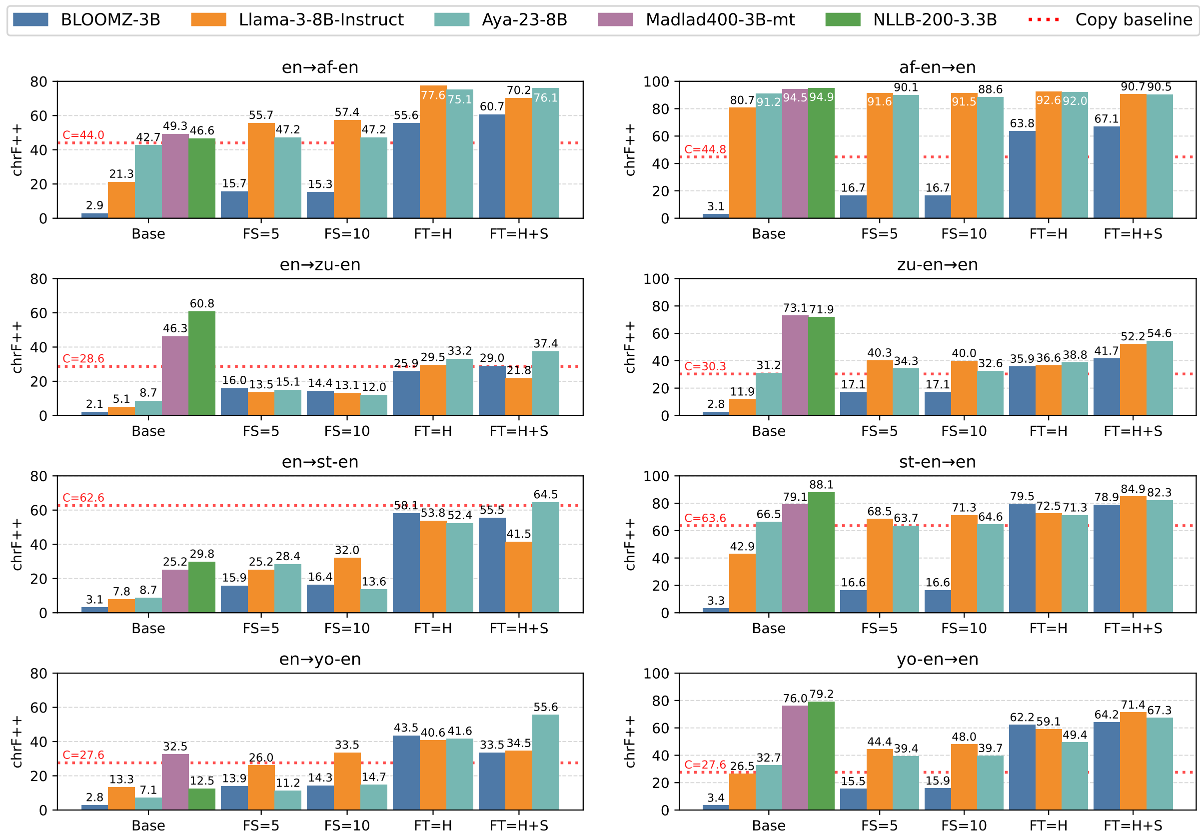

Figure 2: chrF++ scores for the English to code-switch translation task (FS results averaged over 3 seeds).

Figure 3: chrF++ scores for the code-switch to English translation task (FS results averaged over 3 seeds).

sidering the strengths of MT models in translation to English, this task generally produces relatively high chrF++ scores.

MT-only models NLLB-200-3.3B and MADLAD-400-3B-MT consistently outperform LLMs across all settings and languages, demonstrating their efficiency in processing code-switched inputs and leveraging their prior multilingual training. Despite this, LLMs manage to surpass the copy baseline in almost all cases, indicating their ability to generate useful translations without modifying the code-switched English segments in the source text.

In their base configurations, LLMs tend to struggle, particularly with handling the English segments. However, few-shot prompting and fine-tuning with human-validated data (FT=H) significantly enhance their performance. The relatively high scores for Sesotho–English translations across all models may be attributed to the sparse distribution of code-switching ratios, making the task more straightforward compared to other languages.

These findings demonstrate that while LLMs benefit considerably from few-shot prompting and fine-tuning, large MT-only models maintain a clear advantage in translating code-switched text into English, particularly in tasks where maintaining the integrity of pre-existing English content is crucial.[8]

5.3 Further quantitative analysis

Translating natural code-switched Sesotho–English To evaluate the capabilities of LLMs fine-tuned using the AfroCS-xs dataset, we employ a natural code-switched test set of 250 Sesotho-English sentences also on agriculture, manually curated by a native Sesotho speaker (see Section 3.2). The best-performing settings from previous experiments were used for translation. Results are depicted in Figure 4. As anticipated, the scores for both translation directions are generally lower than those from earlier test sets, reflecting the more complex and naturally occurring code-switching patterns not fully captured by the synthetic data in AfroCS-xs. Differences in code-switch ratio and POS distribution, such as a higher presence of ad-

[8]BLEU scores, detailed in Figures 8 and 9 in the Appendix, show similar patterns to chrF++ scores. The main difference is the larger gap between models and the copy baseline in most cases, attributed to BLEU's sensitivity to exact word matches.

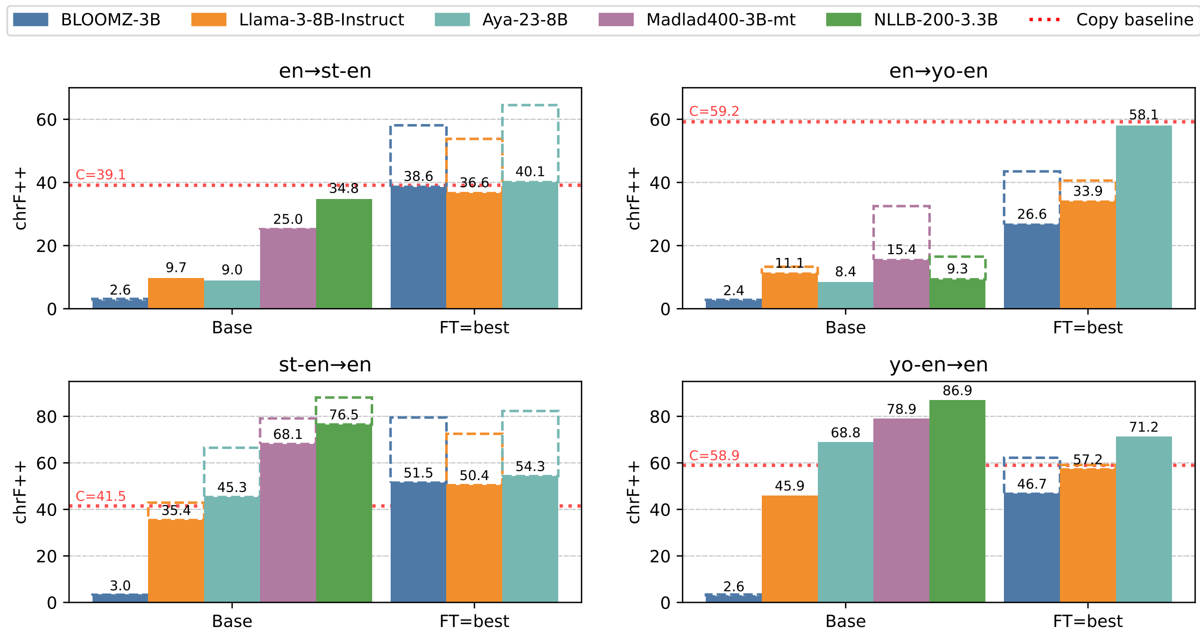

Figure 4: chrF++ scores for the translation task using the real code-switched Sesotho set. Dashed area shows the score in the AfroCS-xs test set. FT=best is the best fine-tune setting per model (FT=H for BLOOMZ-3B & Llama-3-8B-Instruct; FT=H+S for Aya-23-8B).

Figure 5: chrF++ scores for the translation task using the out-of-domain, real code-switched Yoruba set. Dashed area shows the score in the AfroCS-xs test set. FT=best is the best fine-tune setting per model (FT=H for BLOOMZ-3B, Llama-3-8B-Instruct & Aya-23-8B).

jectives, may contribute to this challenge.[9] Despite these complexities, LLMs fine-tuned on AfroCS-xs exhibit notable robustness, with Aya-23-8B showing particularly strong performance in the en→st–en direction. This underscores the adaptability of models fine-tuned on high-quality synthetic data to handle real-world code-switched inputs.

Translating out-of-domain natural code-switched Yoruba-English In a similar vein, we explore the performance of LLMs on the out-of-domain natural code-switched Yoruba–English test set of 334 sentences on personal finance described in Section 3.2. The results, shown in Figure 5, indicate that LLMs consistently surpass the MT baselines in the EN2CS translation direction, demonstrating their ability to generate meaningful code-mixing in underrepresented languages despite low prior exposure. Differences in code-switch ratio and POS distribution, with more adjectives present, along with out-of-distribution terminology, pose additional challenges for generalisation.

Summary The last two experiments further emphasise the importance of diverse and high-quality training data in enhancing model performance across domains and linguistic contexts. Despite the challenges with respect to the copy baseline in both language pairs, our results still underscore that creating a small yet high-quality human-validated code-switch dataset can be sufficient to leverage LLMs beyond other baselines, especially in the EN2CS translation direction task. We also argue that LLMs provide more flexibility than MT models for fine-tuning on very small amounts of curated data, which is typically done to bridge the gap for languages (or domains) MT systems and LLMs are underexposed to.

5.4 Qualitative analysis

Table 8 in Appendix F presents some sample predictions on the Yoruba–English translations from and to English, and briefly highlights certain failure and success modes of the best settings for BLOOMZ-3B[10] and NLLB-200-3.3B.

yo–en→en NLLB-200-3.3B consistently produces English translations that align well with the

[9]As discussed in Section 3.3, the AfroCS-xs data misses a few idiosyncratic patterns observed in natural code-switched data (e.g., different M-index values, slightly different burstiness), thus creating a small shift that impacts the performance of fine-tuned models tested on natural data.

[10]We analyse BLOOMZ's outputs as it is the model with the largest improvement from its base version.

target translation. For example, in the first paragraph of Table 8, the target translation and the translation produced by NLLB-200-3.3B both contain the words *check, suitable* whereas BLOOMZ-3B instead uses the words *monitor, healthy*, which are semantically relevant to the corresponding words *Ṣayẹwo* and *okay* in the source sentence but are not the exact words found in the target translation. BLOOMZ-3B sometimes produces concise translations (see third paragraph in Table 8), and while such translations receive lower chrF++ and BLEU scores compared to NLLB-200-3.3B, they capture the main idea of the source sentences.

en→yo–en The last two paragraphs of Table 8 present some examples from the en→yo–en task. Notably, the fine-tuned BLOOMZ-3B model appears to have learned to incorporate the tone mark style that aligns closely with the style adopted in its fine-tuning set (AfroCS-xs), where only the accent below the alphabets is added for simplicity. In comparison, NLLB-200-3.3B consistently ignores the tone patterns and defaults to the full diacritic markings found in its pre-training data. Additionally, NLLB-200-3.3B frequently translated sentences fully into Yoruba rather than maintaining the code-switched format. These behaviors were also noted in the attempted fine-tunings of NLLB-200-3.3B and MADLAD-400-3B-MT models with AfroCS-xs, with an overall performance not significantly different from that of the base models (see Section 4.3). Overall, the fine-tuned BLOOMZ-3B model produces better quality translations of English into Yoruba–English code-switched sentences than the MT-only models.

6 Conclusions and Future Work

This study introduced AfroCS-xs, a high-quality, human-validated synthetic code-switched dataset for four African languages (Afrikaans, Sesotho, Yoruba, isiZulu) and English for the agricultural domain. By leveraging LLMs, we generated and rigorously validated code-switched sentences and their English translations, providing a valuable resource for under-represented languages. Using this dataset, we fine-tuned various LLMs for code-switch translation tasks and compared their performance against specialist MT-only models.

The results demonstrated that all LLMs improved their translation accuracy with minimal fine-tuning, particularly when using human-validated data. However, the pre-trained MT-first models consistently outperformed the LLMs in translating into English—despite the AfroCS-xs data helping LLMs overcome their deficit when translating into code-switched text. Moreover, the impact of synthetic data varied across models and languages, highlighting the importance of data quality in achieving optimal results. The observed variability in translation performance across different languages, especially with LLMs, underscores the inherent challenges in code-switched translation.

Future work should expand these methodologies to other under-represented languages and explore additional domains to further validate and refine these models. Continued efforts to optimise synthetic data generation and incorporate human validation will be key in overcoming the complexities of code-switch translation.

7 Limitations

The scope of code-switching in this work is limited to Yoruba–English, Afrikaans–English, Sesotho–English, and isiZulu–English. It would be beneficial to include more languages to demonstrate the generality of our conclusions. We also use English as the pivot language to translate from or to. There needs to be more research using low resource languages as the pivot as well.

Furthermore, human validation of synthetic data for machine translation, particularly in code-switched contexts involving African languages, offers potential performance improvements but is not without risks. Subjectivity in validation, validator expertise and bias, error propagation, and the potential for overfitting to specific linguistic preferences can introduce human error into the dataset. These factors can lead to inconsistencies, biases, and errors that may misguide the training of models and affect their translation accuracy. We try to mitigate these risks as much as possible with detailed guidelines described in Appendices B, C, E, but eliminating them entirely would necessitate employing a more diverse group of validators, using more validators per data point, and implementing systematic quality checks to ensure the validated dataset's diversity, representativeness, and objectivity.

Finally, we acknowledge limitations in explaining the varying degrees of improvement observed across the LLMs benchmarked on AfroCS-xs. While all LLMs showed benefits from fine-tuning on AfroCS-xs, the extent of these benefits varied, and this variation is hard to unequivocally interpret due to several confounding factors. One key factor could be the prior exposure of the models to one or more of the low-resource African languages included in AfroCS-xs through their pre and post-training data (see Section 4.2). However, differences in model size, architecture, and specific methodologies employed during pre and post-training could also contribute to the observed variations. Ideally, assessing the impact of prior language exposure on code-switch translation would require comparing a model trained with a language against the same model with that language fully ablated from its prior training data. However, this is nearly impossible for most researchers due to the prohibitive cost of retraining models from scratch and the difficulty of isolating underrepresented languages in data. These limitations highlight the need for more transparent and standardized practices in the development and documentation of LLMs, to facilitate more interpretable evaluations.

8 Ethics Statement

Data Generation Our approach specifically incorporates Agricultural terms and concepts in the prompts to GPT-4, aiming to generate a diverse array of sentences that are contextually rich.

Human Evaluation To ensure the generated sentences are not only linguistically accurate but also culturally sensitive and appropriate, we have involved native speakers of Afrikaans, Yoruba, Sesotho and isiZulu to review our data. This validation process is crucial for ensuring that our research outputs respect cultural and social norms. The annotators and evaluators are co-authors of our paper.

Acknowledgments

This research was funded in part by the Faculty Research Awards of J.P. Morgan AI Research. The authors are solely responsible for the contents of the paper and the opinions expressed in this publication do not reflect those of the funding agencies.

Disclaimer

References

Evershed Amuzu and John Singler. 2014. Codeswitching in west africa. *International Journal of Bilingualism*, pages 329–345.

Ramakrishna Appicharla, Kamal Kumar Gupta, Asif Ekbal, and Pushpak Bhattacharyya. 2021. IITP-MT at CALCS2021: English to Hinglish neural machine translation using unsupervised synthetic code-mixed parallel corpus. In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 31–35.

Viraat Aryabumi, John Dang, Dwarak Talupuru, Saurabh Dash, David Cairuz, Hangyu Lin, Bharat Venkitesh, Madeline Smith, Jon Ander Campos, Yi Chern Tan, Kelly Marchisio, Max Bartolo, Sebastian Ruder, Acyr Locatelli, Julia Kreutzer, Nick Frosst, Aidan N. Gomez, Phil Blunsom, Marzieh Fadaee, Ahmet Üstün, and Sara Hooker. 2024. Aya 23: Open weight releases to further multilingual progress.

Ruthanna Barnett, Eva Codó, Eva Duran Eppler, Montserrat Forcadell, Penelope Gardner-Chloros, Roeland Hout, Melissa Moyer, Maria Torras, Maria Turell, Mark Sebba, Marianne Starren, and Sietse Wensing. 2000. The lides coding manual: A document for preparing and analyzing language interaction data version 1.1–july 1999. *International Journal of Bilingualism*.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. 2022. A survey for in-context learning. *arXiv preprint arXiv:2301.00234*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurélien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Rozière, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Grégoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, and et al. 2024. The llama 3 herd of models.

Björn Gambäck. 2014. On measuring the complexity of code-mixing. In *11th International Conference on Natural Language Processing*.

Björn Gambäck and Amitava Das. 2016. Comparing the level of code-switching in corpora. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1850–1855.

Devansh Gautam, Prashant Kodali, Kshitij Gupta, Anmol Goel, Manish Shrivastava, and Ponnurangam Kumaraguru. 2021. CoMeT: Towards code-mixed translation using parallel monolingual sentences. In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 47–55.

K-I Goh and A-L Barabási. 2008. Burstiness and memory in complex systems. *Europhysics Letters*, 81(4):48002.

Abhirut Gupta, Aditya Vavre, and Sunita Sarawagi. 2021. Training data augmentation for code-mixed translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5760–5766.

Gualberto A. Guzman, Jacqueline Serigos, Barbara E. Bullock, and Almeida Jacqueline Toribio. 2016. Simple tools for exploring variation in code-switching for linguists. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pages 12–20.

Gualberto Guzmán, Joseph Ricard, Jacqueline Serigos, Barbara E. Bullock, and Almeida Jacqueline Toribio. 2017. Metrics for Modeling Code-Switching Across Corpora. In *Proc. Interspeech 2017*, pages 67–71.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spacy: Industrial-strength natural language processing in python. *Zenodo*.

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

Ganesh Jawahar, El Moatez Billah Nagoudi, Muhammad Abdul-Mageed, and Laks Lakshmanan, V.S. 2021. Exploring text-to-text transformers for English to Hinglish machine translation with synthetic code-mixing. In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 36–46.

Tom Kocmi, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes. 2021. To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 478–494, Online. Association for Computational Linguistics.

Prashant Kodali, Anmol Goel, Monojit Choudhury, Manish Shrivastava, and Ponnurangam Kumaraguru. 2022. Symcom - syntactic measure of code mixing a study of english-hindi code-mixing. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 472–480. Association for Computational Linguistics.

Sneha Kudugunta, Isaac Rayburn Caswell, Biao Zhang, Xavier Garcia, Derrick Xin, Aditya Kusupati, Romi Stella, Ankur Bapna, and Orhan Firat. 2023. MADLAD-400: A multilingual and document-level large audited dataset. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Young-Jun Lee, Chae-Gyun Lim, and Ho-Jin Choi. 2022. Does GPT-3 generate empathetic dialogues? a novel in-context example selection method and automatic evaluation metric for empathetic dialogue generation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 669–683.

Yongkang Liu, Shi Feng, Daling Wang, and Yifei Zhang. 2022. MulZDG: Multilingual code-switching framework for zero-shot dialogue generation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 648–659. International Committee on Computational Linguistics.

Vukosi Marivate, Tshephisho Sefara, Vongani Chabalala, Keamogetswe Makhaya, Tumisho Mokgonyane, Rethabile Mokoena, and Abiodun Modupe. 2020. Investigating an approach for low resource language dataset creation, curation and classification: Setswana and sepedi. In *Proceedings of the first workshop on Resources for African Indigenous Languages*, pages 15–20.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, et al. 2022. Crosslingual generalization through multitask finetuning. *arXiv preprint arXiv:2211.01786*.

El Moatez Billah Nagoudi, AbdelRahim Elmadany, and Muhammad Abdul-Mageed. 2021. Investigating code-mixed Modern Standard Arabic-Egyptian to English machine translation. In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 56–64.

Wilhelmina Nekoto, Vukosi Marivate, Tshinondiwa Matsila, Timi Fasubaa, Taiwo Fagbohungbe, Solomon Oluwole Akinola, Shamsuddeen Muhammad, Salomon Kabongo Kabenamualu, Salomey Osei, Freshia Sackey, Rubungo Andre Niyongabo, Ricky Macharm, Perez Ogayo, Orevaoghene Ahia, Musie Meressa Berhe, Mofetoluwa Adeyemi, Masabata Mokgesi-Selinga, Lawrence Okegbemi, Laura Martinus, Kolawole Tajudeen, Kevin Degila, Kelechi Ogueji, Kathleen Siminyu, Julia Kreutzer, Jason Webster, Jamiil Toure Ali, Jade Abbott, Iroro Orife, Ignatius Ezeani, Idris Abdulkadir Dangana, Herman Kamper, Hady Elsahar, Goodness Duru, Ghollah Kioko, Murhabazi Espoir, Elan van Biljon, Daniel Whitenack, Christopher Onyefuluchi, Chris Chinenye Emezue, Bonaventure F. P. Dossou, Blessing Sibanda, Blessing Bassey, Ayodele Olabiyi, Arshath Ramkilowan, Alp Öktem, Adewale Akinfaderin, and Abdallah Bashir. 2020. Participatory research for low-resourced machine translation: A case study in African languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2144–2160.

Iroro Orife, Julia Kreutzer, Blessing K. Sibanda, Daniel Whitenack, Kathleen Siminyu, Laura Martinus, Jamiil Toure Ali, Jade Z. Abbott, Vukosi Marivate, Salomon Kabongo KABENAMUALU, Musie Meressa, Espoir Murhabazi, Orevaoghene Ahia, Elan Van Biljon, Arshath Ramkilowan, Adewale Akinfaderin, Alp Oktem, Wole Akin, Ghollah Kioko, Kevin Degila, Herman Kamper, Bonaventure F. P. Dossou, Chris C. Emezue, Kelechi Ogueji, and Abdallah Bashir. 2020. Masakhane - machine translation for africa. *AfricaNLP Workshop, ICLR*.

Shana Poplack. 1980. Sometimes I'll start a sentence in Spanish y termino en español: Toward a typology of codeswitching. *Linguistics*, pages 581–618.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Vikas Raunak, Arul Menezes, and Hany Awadalla. 2023. Dissecting in-context learning of translations in GPT-3. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 866–872.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. Learning to retrieve prompts for in-context learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2655–2671.

Kai Song, Yue Zhang, Heng Yu, Weihua Luo, Kun Wang, and Min Zhang. 2019. Code-switching for enhancing NMT with pre-specified translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 449–459.

NLLB Team, Marta R. Costa-jussà, 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. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Michelle Terblanche, Kayode Olaleye, and Vukosi Marivate. 2024. Prompting towards alleviating code-switched data scarcity in under-resourced languages with GPT as a pivot. In *Proceedings of the 3rd Annual Meeting of the Special Interest Group on Under-resourced Languages @ LREC-COLING 2024*, pages 272–282.

Maarten van Gompel, Iris Hendrickx, Antal van den Bosch, Els Lefever, and Véronique Hoste. 2014. SemEval 2014 task 5 - L2 writing assistant. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 36–44.

Zhen Wan, Fei Cheng, Zhuoyuan Mao, Qianying Liu, Haiyue Song, Jiwei Li, and Sadao Kurohashi. 2023. GPT-RE: In-context learning for relation extraction using large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3534–3547.

Jiayi Wang, David Ifeoluwa Adelani, Sweta Agrawal, Marek Masiak, Ricardo Rei, Eleftheria Briakou, Marine Carpuat, Xuanli He, Sofia Bourhim, Andiswa Bukula, Muhidin Mohamed, Temitayo Olatoye, Tosin Adewumi, Hamam Mokayed, Christine

Mwase, Wangui Kimotho, Foutse Yuehgoh, Anuoluwapo Aremu, Jessica Ojo, Shamsuddeen Hassan Muhammad, Salomey Osei, Abdul-Hakeem Omotayo, Chiamaka Chukwuneke, Perez Ogayo, Oumaima Hourrane, Salma El Anigri, Lolwethu Ndolela, Thabiso Mangwana, Shafie Abdi Mohamed, Hassan Ayinde, Oluwabusayo Olufunke Awoyomi, Lama Alkhaled, Sana Al-azzawi, Naome A. Etori, Millicent Ochieng, Clemencia Siro, Njoroge Kiragu, Eric Muchiri, Wangari Kimotho, Lyse Naomi Wamba Momo, Daud Abolade, Simbiat Ajao, Iyanuoluwa Shode, Ricky Macharm, Ruqayya Nasir Iro, Saheed S. Abdullahi, Stephen E. Moore, Bernard Opoku, Zainab Akinjobi, Abeeb Afolabi, Nnaemeka Obiefuna, Onyekachi Raphael Ogbu, Sam Ochieng', Verrah Akinyi Otiende, Chinedu Emmanuel Mbonu, Sakayo Toadoum Sari, Yao Lu, and Pontus Stenetorp. 2024. AfriMTE and AfriCOMET: Enhancing COMET to embrace under-resourced African languages. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5997–6023, Mexico City, Mexico. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing. *Preprint*, arXiv:1910.03771.

Jitao Xu and François Yvon. 2021. Can you traducir this? machine translation for code-switched input. In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 84–94.

Zheng Xin Yong, Ruochen Zhang, Jessica Forde, Skyler Wang, Arjun Subramonian, Holy Lovenia, Samuel Cahyawijaya, Genta Winata, Lintang Sutawika, Jan Christian Blaise Cruz, Yin Lin Tan, Long Phan, Long Phan, Rowena Garcia, Thamar Solorio, and Alham Aji. 2023. Prompting multilingual large language models to generate code-mixed texts: The case of south East Asian languages. In *Proceedings of the 6th Workshop on Computational Approaches to Linguistic Code-Switching*, pages 43–63.

A Prompt Templates for Generating AfroCS-xs via GPT-4

Prompt A You are a game developer who wants to build an interactive and educational Farming game. Part of the game involves communicating with players in their local language for example, if a player is a rural farmer who code-switches, the instructions should be provided to the player in `{code-switch language}` with `{matrix language}` as the the matrix language and English as the embedded language. The purpose of the game is to entertain and teach the player how to run or start a crop farm from scratch. Let's start by providing 5 code-switched sentences relevant to get a player started on the game platform.
Prompt B You are a game developer who wants to build an interactive and educational game about Agriculture and farming for youths who don't necessarily have an interest in Agriculture and farming. You want the game to be culturally compliant. So if you are building the game for Nigerian Youths, you want the game to communicate and interact with the player in Nigeria's local languages. For example, if a player is a rural farmer who code-switches, every in-game text should be provided to the player in `{code-switch language}` with `{matrix language}` as the matrix language and English as the embedded language. Here are some examples: Vocabularies: `{agriculture-related keywords}` `{5-shot code-switched sentences}` So now I will give words in English and then you generate `{code-switch language}` sentences relevant to different pieces of the game from teaching the player about crop farming, explaining concepts, providing instructions etc. Your knowledge should include everything you know about running a poultry business from scratch. Vocabularies: `{new agriculture-related keywords}` Give 30 Examples of code-switch sentences (`{matrix language}` as the matrix language and English as the embedded language). Each sentence should have a minimum of 10-15 words. Don't put the words in front, don't give the English translations, remove any sentence that does not contain code-switching, and ensure the sentences are culturally appropriate.
Prompt C Give the English translations

Table 4: Prompt templates for generating 5-shot examples (Prompt A), generating the code-switched sentences (Prompt B), and generating English translations (Prompt C)

Synthesising code-switched sentences in one language using examplars from another In our study, we also explore the cross-linguistic generation of code-switched sentences, a process wherein few-shot examples of code-switching in one language pair are employed to guide GPT-4 in generating code-switched sentences in another target language pair. For instance, we synthesise Sesotho–English code-switched sentences using Yoruba–English code-switched examples as the model's directive. This approach tests the capacity of language models to transfer code-switching patterns from one language pair (Yoruba–English) to another (Sesotho–English), despite the lack of direct few-shot examples in Sesotho–English. The specifics of the prompt used for this task can be found in Prompt D, detailed in Table 5. This methodology not only assesses the adaptability of language models to navigate between diverse linguistic structures but also their potential to facilitate multilingual communication in novel, resource-scarce settings.

Prompt D You are a game developer who wants to build an interactive and educational game about Agriculture and farming for youths who don't necessarily have an interest in Agriculture and farming. You want the game to be culturally compliant. So if you are building the game for Nigerian Youths, you want the game to communicate and interact with the player in Nigeria's local languages. For example, if a player is a rural farmer who code-switches, every in-game text should be provided to the player in `{code-switch language1}` with `{matrix language1}` as the matrix language and English as the embedded language. Here are some examples: Vocabularies: `{agriculture-related keywords in matrix language1}` `{5-shot code-switched sentences in language pair1}` So now I will give words in English and then you generate `{code-switch language2}` sentences relevant to different pieces of the game from teaching the player about crop farming, explaining concepts, providing instructions etc. Your knowledge should include everything you know about running a poultry business from scratch. Vocabularies: `{new agriculture-related keywords in matrix language2}` Give 30 Examples of code-switch sentences (`{matrix language2}` as the matrix language and English as the embedded language). Each sentence should have a minimum of 10-15 words. Don't put the words in front, don't give the English translations, remove any sentence that does not contain code-switching, and ensure the sentences are culturally appropriate.

Table 5: The prompt template for generating code-switched sentences without direct few-shot examples from the same matrix language.

B Annotation Guidelines for Code-Switched Sentences

The annotation process comprises three phases:

1. Correction of code-mixed sentences
2. POS-tagging of corrected sentences
3. Word-level language identification of corrected sentences

Note: This version of the guidelines focuses on Phase 1.

Phase 1: correction of code-switched sentences
Objective:

- Refine code-mixed sentences to ensure they represent a natural, culturally relevant blend of the language pair.

Instructions:

- Authenticity: Adjust sentences to mirror natural code-mixing patterns observed among native speakers.
- Cultural Relevance: Ensure that the corrected sentences resonate with the cultural context of the language pair.
- Structural Flexibility: You may restructure sentences to better reflect typical code-switching dynamics.

- Tonal Languages: For tonal languages, diacritics are optional in this phase. However, maintain any under-dots present.

- Preservation of Quality: If the original sentence is already well-formed, do not alter it.

Additional Guidelines:

- Commenting on Corrections: Use the comment column in the Google Sheet to note observations or comments about sentences that require minor or major corrections.

- No Changes Made: If no changes are made to a sentence, leave the comment column blank.

C Guidelines for Human Qualitative Analysis of Models' Translation Performance

You will be given access to a Google Drive folder. In the folder, you will find a list of Google Sheets. You will find in each Google Sheet the models' translations of

- Code-switched sentences to English sentences. You will find translations produced by models fine-tuned on synthetic data and those produced by models fine-tuned on human-validated synthetic data.

- English sentences to code-switched sentences.

I will try to highlight (in green) the sentences which you will qualitatively evaluate/interpret—you don't have to analyse all rows in each Google Sheet. The goal is to provide a concise yet comprehensive qualitative analysis of the translations in a single paragraph. In the paragraph, you can comment on the following:

- Start by assessing the overall accuracy of the translation. Mention how well the translation captures the original text's meaning, including nuances and idiomatic expressions.

- Evaluate the fluency and naturalness of the translation. Consider whether it reads as if originally written in the target language, and note any awkward or unnatural phrases.

- Discuss the translation's contextual understanding. Does it effectively convey the original text's tone, style, and emotional undercurrents?

- Comment on cultural appropriateness. Observe if the translation respects cultural nuances and sensitivities.

- Mention the type of mistakes the models make while translating.

- Note the consistency in terminology, and style throughout the translation.

- Conclude with your overall impression of the translation's effectiveness in communicating the intended message to the target audience.

D Model Sensitivity to Prompt Variations

Table 7 shows that there is minimal difference in the performance of the BLOOMZ-3B model when fine-tuned using a more descriptive template (Prompt I) versus a simpler one (Prompt II) (See Table 6 for the definition of Prompts I and II). The BLEU score for various translation tasks, under conditions where the fine-tuning is done with either 15 or 60 examples (code-switched sentences and their corresponding English translations), indicates only slight variations. For instance, the zu–en $\rightarrow$ en translation shows only modest increases, with BLEU scores increasing from 12% to 20% for Prompt I and from 13% to 21% for Prompt II.

Prompt I - CS2EN

I want you to act as a translator. Below is a code-mixed sentence in `{language pair}`. *Please write the English translation.*

`{language pair}`: `{code switched sentence}`

English: `{english translation}`

Prompt II - CS2EN

`{language pair}`: `{code switched sentence}`

English: `{english translation}`

Table 6: The prompt templates for transforming parallel English and code-switched texts into a prompt.

		af–en → en		zu–en → en		st–en → en		yo–en → en	
		15	60	15	60	15	60	15	60
1	Prompt I	42.0	47.0	12.0	20.0	61.0	69.0	23.0	46.0
2	Prompt II	43.0	51.0	13.0	21.0	60.0	71.0	29.0	42.0

Table 7: Comparison of BLEU scores for code-switched translation tasks under different prompting templates and fine-tuning conditions (15 and 60 examples).

E Creating the Natural Code-Switched Sentences for Further Evaluation of Model Performance

E.1 Natural Sesotho–English code-switched sentences

We provided a native Sesotho speaker (the annotator) with 250 synthetically generated English sentences covering different topics in agriculture. The annotator is fluent in both the English and the Sesotho language and can code-switch using the two languages. The following instructions were provided to the annotator to complete the code-switched data creation task:
Rewrite each English sentence in Sesotho-English code-switched form. Ensure the resulting sentences adheres to the natural code-switching patterns observed among native speakers. The resulting sentences should be a good blend of Sesotho and English. You are welcome to rewrite as many sentences as your schedule allows.

Feedback from the Annotator regarding the task. The sentences were initially written in English to facilitate code-switching into Southern Sesotho. To achieve this, each sentence was read three times to ensure comprehension. However, due to challenging scientific vocabulary, we consulted the Oxford English Dictionary and online resources for definitions. After understanding these terms, we reread the sentences twice more to confirm their meanings. Next, we translated the entire sentences into Southern Sesotho, selectively replacing certain words while retaining others in English. For scientific terms without direct Southern Sesotho equivalents, we applied prefixes and suffixes, as indigenous African languages often incorporate these in their word structures. This process involved multiple readings to ensure the sentences retained their intended meaning, taking up to 24 hours or a full day to complete. Although the task was tedious, it was also enjoyable. The primary challenge was the lack of Southern Sesotho or other indigenous African language equivalents for many scientific terms, necessitating the use of prefixes and suffixes. Interestingly, applying these linguistic elements to scientific words made the text easier to read and understand. The resulting sentences, combining Southern Sesotho and English, create a unique form of urban language. This code-switching is surprisingly comprehensible for speakers of various Sesotho languages, such as Sesotho, Setswana, and Sepedi, which share a similar orthographic structure.

E.2 Natural and out-of-domain Yoruba–English code-switched sentences

We provided a native Yoruba speaker (the annotator) with 300 synthetically generated English sentences covering various topics in personal finance. The annotator, fluent in both English and Yoruba, was instructed to code-switch these sentences using the two languages. The task guidelines were as follows:
Rewrite each English sentence in Yoruba-English code-switched form, ensuring that the resulting sentences

adhere to natural code-switching patterns observed among native speakers. The sentences should effectively blend Yoruba and English, maintaining the integrity of the original content while incorporating Yoruba expressions.

Feedback from the Annotator regarding the task. We spent a total of 18 hours completing the task, during which we often referred to a dictionary to accurately translate some of the more complex terms into Yoruba. Initially, we encountered considerable difficulty, especially with understanding and translating specific financial terminology. However, as the project progressed, we found the task increasingly enjoyable and engaging. Part of our process involved saying each sentence out loud to see how well it mimicked a pattern similar to real-life usage. This oral review helped ensure that the Yoruba-English code-switched sentences sounded natural and authentic, closely resembling the conversational dynamics observed among native speakers. Additionally, we took care to correct any typographical errors and removed any redundant sentences to avoid duplicative work. This reflective and iterative approach not only ensured the quality and accuracy of the final output but also enhanced our overall satisfaction and educational experience.

F Complementary Results

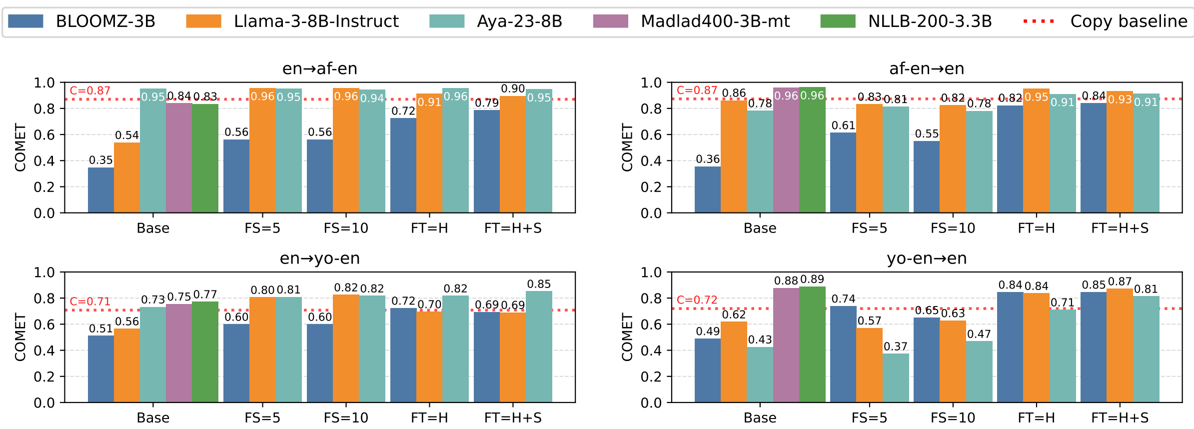

Figure 6: COMET scores for the English to code-switch translation task. The scoring models are wmt22-comet-da (Rei et al., 2020) and AfriCOMET-MTL (Wang et al., 2024) for Afrikaans and Yoruba, respectively. Sesotho and isiZulu are not supported.

Figure 7: COMET scores for the code-switch to English translation task. The scoring models are wmt22-comet-da (Rei et al., 2020) and AfriCOMET-MTL (Wang et al., 2024) for Afrikaans and Yoruba, respectively. Sesotho and isiZulu are not supported.

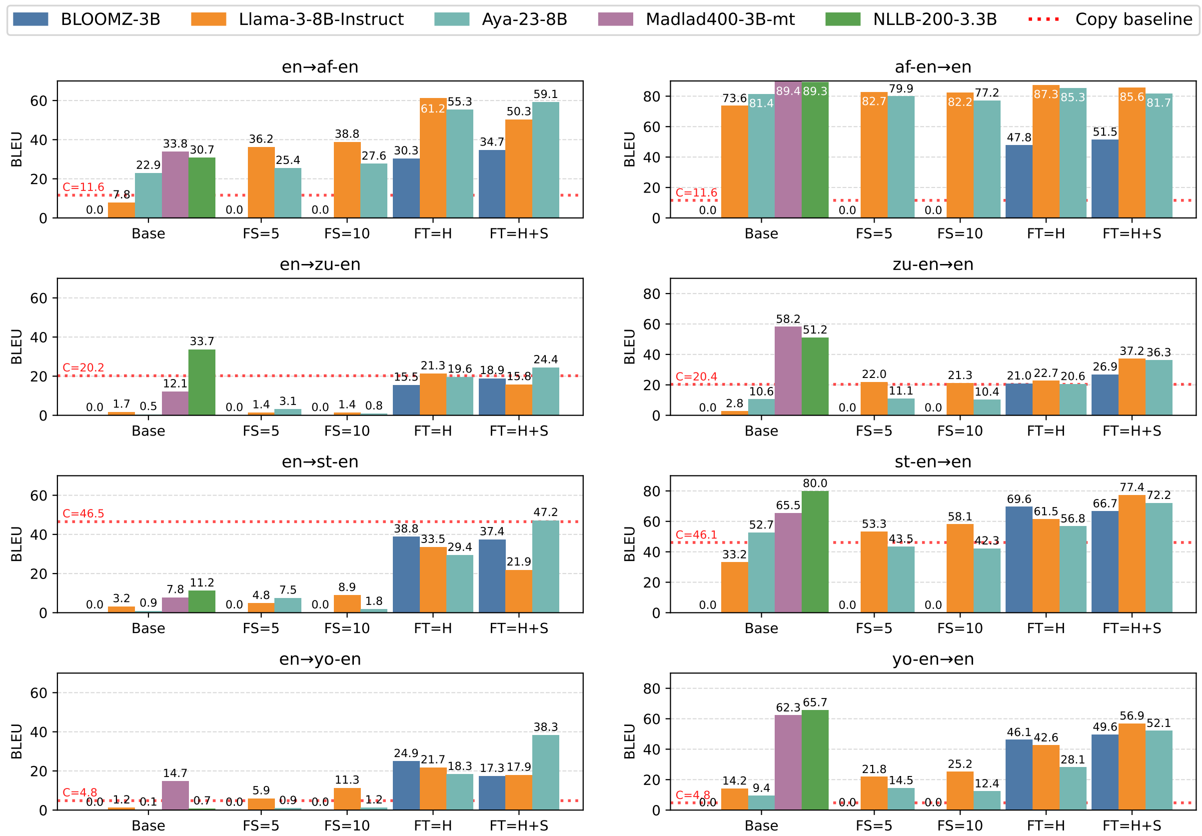

Figure 8: BLEU scores for the English to code-switch translation task (FS results averaged over 3 seeds).

Figure 9: BLEU scores for the code-switch to English translation task (FS results averaged over 3 seeds).

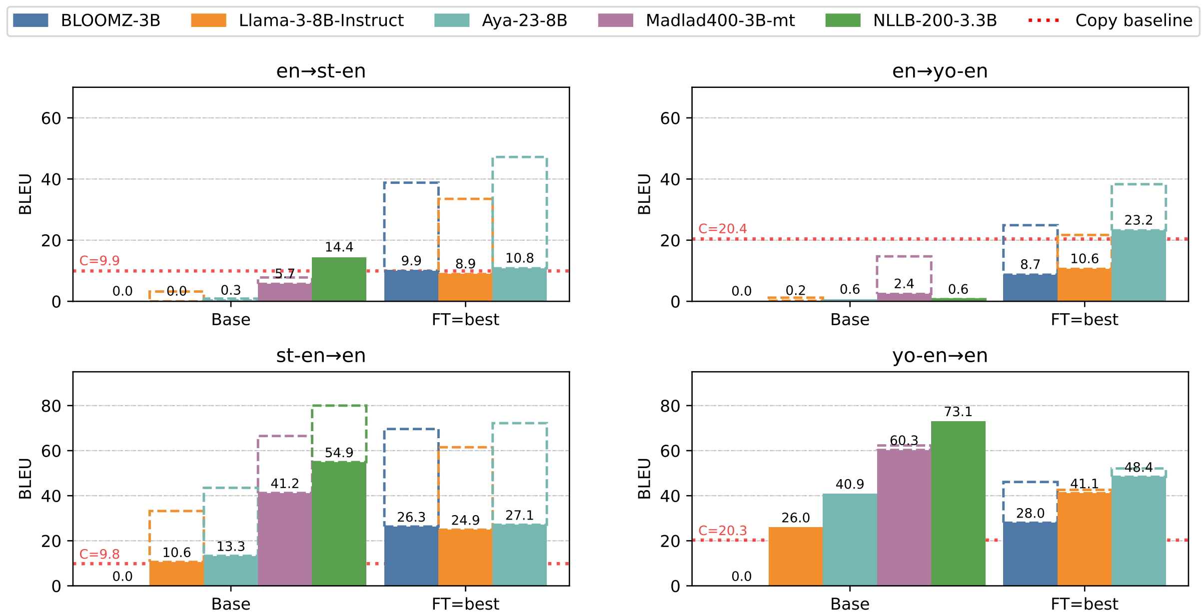

Figure 10: BLEU scores for the translation task using the real code-switched Sesotho set. Dashed area shows the score in the AfroCS-xs test set. FT=best is the best fine-tune setting per model (FT=H for BLOOMZ-3B & Llama-3-8B-Instruct; FT=H+S for Aya-23-8B).

Figure 11: BLEU scores for the translation task using the out-of-domain and real code-switched Yoruba set. Dashed area shows the score in the AfroCS-xs test set. FT=best is the best fine-tune setting per model (FT=H for BLOOMZ-3B, Llama-3-8B-Instruct & Aya-23-8B).

Task: yo-en $\rightarrow$ en
Source: *Ṣayẹwo pH soil ki o to gbin millet, lati rii daju pe o wa okay.*
Target: *Check the pH of the soil before planting millet, to ensure it's suitable.*
BLOOMZ-3B (h): *Monitor the soil's pH before planting millet, to ensure that it is healthy.*
NLLB-200-3.3B (base): *Check the pH of the soil before planting millet, to ensure it is suitable.*

Source: *Yan awọn irugbin rice ti o le koju aisan ati local climate agbegbe rẹ.*
Target: *Choose rice seeds that can resist diseases and suit your local climate.*
BLOOMZ-3B (h): *Avoid planting crops that are susceptible to pests and diseases of the area.*
NLLB-200-3.3B (base): *Choose rice crops that are resistant to disease and local climate in your area.*

Source: *Mimọ nipa pest control jẹ pataki lati daabo bo wheat rẹ lọwọ awọn kokoro.*
Target: *Knowing about pest control is crucial to protect your wheat from insects.*
BLOOMZ-3B (h): *A knowledge of pest control is important to prevent diseases in wheat.*
NLLB-200-3.3B (base): *Knowledge of pest control is essential to protect your wheat from germs.*

Task: en $\rightarrow$ yo-en
Source: *Currently, you need to learn how to deal with mistakes.*
Target: *Lọwọlọwọ, o nilo lati kọ ẹkọ bi a ṣe le deal pẹlu awọn aṣiṣe.*
BLOOMZ-3B (h): *Ẹ gbọdọ kọ bi a ṣe le deal with mistakes.*
NLLB-200-3.3B (base): *Ní bàyìí, o gbọ́dọ̀ kọ́ bó ṣe lé borí àwọn àṣìṣe rẹ.*

Source: *Rice can grow well in water-rich areas, so check your irrigation system.*
Target: *Rice le grow daadaa ni agbegbe ti o ni omi to, nitorina ṣayẹwo eto irrigation rẹ.*
BLOOMZ-3B (h): *Rice le grow well ni agbado ti o ni omi pupọ, nitorina ṣayẹwo eto irrigation rẹ.*

Source: *To grow millet well, you must use effective irrigation techniques.*
Target: *Lati grow millet daradara, o gbọdọ lo awọn ilana irrigation ti o munadoko.*
BLOOMZ-3B (h): *Lati grow millet well, o gbọdọ lo effective irrigation techniques.*

Table 8: Translations to and from English produced by BLOOMZ-3B and NLLB-200-3.3B models. 'h' indicates models fine-tuned with human-validated data, 'h + s' indicates models fine-tuned with human-validated and unvalidated data. 'base' indicates using the model without fine-tuning.