Dataset Extraction and Creation for Social Services in Africa

Anonymous Author(s)
Affiliation
Address
email

Abstract

Large Language Models (LLMs) are transforming digital services globally, yet their integration into localized and resource-constrained environments, such as those in Africa, remains underexplored. This paper presents an extensive approach from dataset creation to real-world model deployment of fine-tuned LLMs including Llama, Gemma and Deepseek for structured financial, healthcare and communication services. We develop structured datasets tailored to African contexts, fine-tune several open-source models, and evaluate their ability to accurately extract key details from informal messages into structured JSON outputs. We integrate the best performing model into our already existing WhatsApp-based AI assistant capable of performing tasks like sending reminders, scheduling payments and providing healthcare reminders. A comparative analysis reveals the differences in model performances, highlighting the best approaches for efficient deployment in resource-limited African markets. Our findings suggest that LLM-based solutions are viable in bridging the gap in digital services in low-resource settings, allowing for inclusivity and accessibility.

1 Introduction

Large Language Models (LLMs) are being leveraged all around the globe, offering new opportunities to drive innovation and improve efficiency in and around various industries and sectors, transforming the way different businesses operate. These models have indeed demonstrated exceptional capabilities across a range of tasks including machine translation and summarization in even some low-resource contexts as demonstrated in research.

In the context of structured digital services, high-resource settings have made great strides due to the availability of large-scale data making the possibility of creating such applications easy, as compared to low-resource contexts where there are still limitations on the deployment of such models, owing to the scarce standardized data. There is a growing need for AI systems that are context-aware and task-specific in African markets where informal communication norms still exist and mobile first interactions dominate.

Social services in areas such as financial planning, healthcare, and public administration are critical for communities across Africa. AI technologies have the potential to enhance these services by providing intelligent support and decision-making, from automating appointment scheduling to offering personalized health advice. Indeed, early applications of AI in African contexts have shown promise: for example, machine learning on satellite imagery has been used to estimate poverty levels and support financial inclusion efforts[Jean et al., 2016], and digital data have been harnessed for public health surveillance[Zhao et al., 2020]. These successes illustrate the opportunities of AI-for-social-good in emerging markets. However, mainstream large language models (LLMs) and AI systems often underperform or lack support for African languages and local contexts[Chen et al.,

Submitted to 7th Deep Learning Indaba Conference (DLI 2025). Do not distribute.

2024]. A recent study evaluating ChatGPT (powered by GPT-4) on 670 languages found that it struggles the most with African languages[Chen et al., 2024], recognizing text in Hausa only about 10–20% of the time[Moorosi, 2024]. Such disparities stem from a shortage of high-quality, relevant training data for low-resource languages and domains[Moorosi, 2024].

There is a pressing need to create datasets that reflect African social service scenarios and languages. Traditional dataset curation (e.g., manually collecting conversational data in multiple local languages) is expensive and time-consuming. As an alternative, we explore using LLMs to *generate* and *extract* structured data that can bootstrap model training. Recent advances show that with the right prompts, LLMs can produce outputs in structured formats like JSON or XML [Shorten et al., 2024]. By guiding an LLM to output, for instance, a JSON record of a dialogue or a service log, we can automatically create synthetic training examples. This approach leverages the generative power of LLMs to overcome data scarcity, effectively turning the model into a data generator for downstream tasks.

In this paper, we present a framework for designing, fine-tuning and deploying LLMs tailored to structured social services in Africa, such as financial reminders, scheduling, and healthcare-related communication, using datasets from African contexts derived from real-world use cases.

Our contribution will include the following:

- Construction of task-specific datasets that reflect the low-resource scenarios, and annotated for structured outputs.
- A fine-tuning and evaluation pipeline applied to open-source models (Llama, Gemma, and Deepseek), and adapted to perform structured JSON generation from natural prompts.
- A benchmarking methodology using confidence-based sampling to assess the model performance.

2 Related Work

2.1 Structured Generation from LLMs

Structured output generations have recently been an area of interest of research for adapting LLMs to domain-specific tasks. Some early methods focused on templated data or classification heads and earlier work like T5 and GPT-3 explored generation of machine-readable outputs using prompt engineering and task formulations, however more recent work has explored direct generation of JSON or XML structures. For example, [Geng et al., 2025] introduced JSONSchemaBench, which evaluates LLMs on their ability to follow predefined schemas during generation. [Lu et al., 2025] proposed schema-aware reinforcement learning to guide structured generation.

Shorten et al. [2024] introduce a benchmark called StructuredRAG to assess how well LLMs follow response formatting instructions, reflecting growing interest in reliable structured generation. They report that state-of-the-art models can achieve over 80% success in zero-shot structured output tasks, though performance varies by task complexity. Other approaches enforce structure via constrained decoding algorithms [Beurer-Kellner et al., 2024], ensuring, for example, that parentheses or JSON brackets are balanced and keys are present. Tam et al. [2024] find that while models can be coaxed into formats like JSON, overly rigid format requirements may sometimes degrade the quality of the answer.

These approaches emphasize fine-tuning methods, Retrieval-Augmented Generations (RAGs), and prompt-engineering to ensure synthetic and semantic validity. Our approach falls within this category, but we emphasize task-specific fine-tuning from real-world service messages. These efforts highlight the promise of LLMs in producing structured formats, but they are often benchmarked on synthetic tasks. Our work grounds this approach in realistic, service-driven datasets.

2.2 Conversational Interface for Social Services

There has been a recent increasing interest in using LLMs for the support of public service delivery. A number of researchers have explored using language models for public-facing tasks in health and finance. [Singhal et al., 2022] developed Med-PaLM, a model fine-tuned on medical question answering, while [Rakesh et al., 2025] introduced ChatFin, an LLM-based banking assistant capable

of processing finance-related chat inputs. However, these models typically rely on few-shot prompting or in-context learning, rather than full finetuning.

Deploying such AI solutions in practice often requires conversational interfaces and local language support, especially for outreach and education. For instance, a health advisory system might take symptoms described by a user in Swahili and provide guidance or triage information. Similarly, a financial planning assistant might converse in French with users in Francophone African countries to help them budget or access micro-loans. These use cases demand both the underlying predictive analytics and a user-facing conversational component.

Our work complements this by developing a fine-tuned model for narrow, operational tasks such as reminders and scheduling which require precise extraction and generation.

2.3 NLP for Low-Resource Context

Majority of the literature in low-resource context NLP focuses on multilingual modeling, transfer learning (Hedderich et al. [2021]) as well as dataset curation. These approaches have been aimed to extend language technologies to low-resource settings. Our work, in addition to creating datasets, is situated in a setting where our notion of low-resource refers not to the language per se, but to the availability of aligned task-specific high quality data, compute and automation infrastructure.

3 Methodology

Our approach combines fine-tuning large language models (LLMs) with structured information retrieval to enhance AI-driven services in low-resource environments. We focus on financial services, healthcare, and general task reminders in English and French, reflecting real-world usage patterns in West and Central Africa.

3.1 Dataset Construction

Our approach to generating the dataset involved curating six different dataset from real-world WhatsApp-style prompts, ranging from financial to healthcare, collected through user experiments. In constructing the dataset, for each data instance, we prompt engineered GPT 4o through the OpenAI Azure Foundry to create an annotated structured JSON completion, retrieving fields from the initial CSV data files such as the sender, recipient, amount, deadlines, actions and other fields. This further went through manual annotation for correction. We then performed some additional normalization to the phone format and date values.

Example Dataset

Below are two examples of prompt-completion pairs used for fine-tuning. Each example consists of a user-provided instruction and the expected structured output.

```
{
  "prompt": "Hi, could you send a reminder to 96 28 81 35 for a debt of 41243.13
  ↪  FCFA due on 03/05/2025? Reminder to be sent on 01/08/2025.",
  "completion": {
    "sender_phone": "N/A",
    "recipient_phone": "96 28 81 35",
    "action": "send reminders",
    "action_count": 1,
    "deadline_date": "2025-05-03",
    "reminder_date": "2025-08-01",
    "frequency": "none",
    "response-to-sender": "The reminder for the debt of 41243.13 FCFA due on
    ↪  2025-05-03 will be sent to 96 28 81 35 on 2025-08-01.",
    "response_to_recipient": "You have a debt of 41243.13 FCFA due on 2025-05-03.
    ↪  This is a reminder sent as requested."
  }
}
```

We then merge and clean the datasets, creating a training corpus of approximately **8700** examples and a benchmark test set, which we constructed via a confidence-based sampling approach and manual verification.

3.2 Sampling Dataset for Benchmarking

We follow a statistical approach which uses the confidence interval calculation. We calculate the initial sample size using the z-score from the confidence interval chosen, the data size, an estimated proportion and a margin of error. Once that is done, we use the initial sample size to adjust for a finite population, which is then going to be our required sampled size.

The sample size calculation employs the **finite population correction** formula, suitable for smaller datasets, given by:

$$n = \frac{n_0}{1 + \frac{(n_0 - 1)}{N}}, \tag{1}$$

where the initial sample size n_0 is calculated as:

$$n_0 = \frac{Z^2 \times p \times (1 - p)}{E^2}. \tag{2}$$

Parameters Explained:

- N: Population size (total records in the dataset)
- Z: Z-score (e.g., $Z = 1.96$ for 95% confidence)
- p: Estimated proportion
- E: Margin of error

3.3 Fine-tuning Setup

We finetuned three open-source LLMs using task-specific LoRA adapters:

- LLaMA (1B)
- DeepSeek (1.5B)
- Gemma (2B)

Models were trained using Unsloth's optimized trainer with 4-bit quantization. We used a ChatML-style prompt format and tuned models on JSON generation tasks using SFT (supervised finetuning).

3.4 Evaluation Metrics

We assessed the performance of the finetuned models using both exact and partial matching metrics, including exact match accuracy, field-level accuracy, and token-level metrics such as BLEU and ROUGE-L (on text responses).

1. **Exact Match Accuracy:** This metric compares the model's prediction string to the reference string to measure how often the strings match(IBM). This metric is strict, where all labels have to match correctly for a correct classification (Lukasik et al. [2024].
2. **Field Accuracy:** To evaluate models more leniently, we compute token-level correctness across all fields. Each field is treated independently.
3. **BLEU:** To assess the quality of free-form text fields—particularly the fields **'response-to-sender'** and **'response-to-recipient'**, we use BLEU (Bilingual Evaluation Understudy). We report the average BLEU over the two response fields.
4. **ROUGE-L F1:** The longest common subsequence between the reference and model's prediction is captured by the ROUGE-L F1 score, which we calculate. ROUGE-L is appropriate for assessing the coherence and fluency of generated responses and places an emphasis on sequence-level similarities.

4 Results and Discussion

The results for the finetuned models are presented in two parts. First, we report the training loss curves for each of the models, and then we evaluate model performance using structured prediction metrics, as mentioned in Subsection 3.4.

4.1 Training Losses

All the models were trained for approximately 4300 steps. The number of epochs varied due to the variation in dataset tokenization and batch length: LLaMA was trained for 2 epochs, while Gemma and DeepSeek were each trained for 8 epochs.

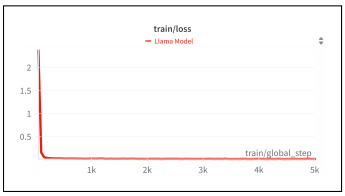

Figure 1: LLaMA Train Loss

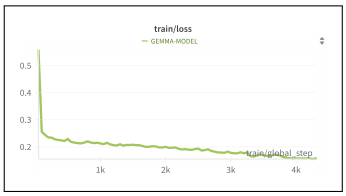

Figure 2: Gemma Train Loss

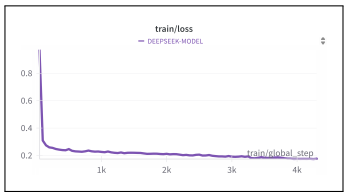

Figure 3: DeepSeek Train Loss

The diagrams in Figure 4.1 above, shows the progression of the losses for three finetuned models (i.e., Llama, Gemma, and Deepseek) on our datasets. DeepSeek decreases the fastest and the most steadily, stabilizing below a loss value of 0.1759. Gemma's loss decreases smoothly to around 0.1577 with minor fluctuations. LLaMA starts from a very different and higher loss value (>2.0), perhaps due to different scaling or initialization, but converges quickly to similar values as the other two models.

4.2 Evaluation Results

Table 1 shows a summary of the evaluation results for the three fine-tuned models on the structured prediction task. Metrics include exact match accuracy, field-level accuracy, BLEU score, and ROUGE-L F1.

Table 1: Evaluation results for the fine-tuned models on structured prediction.

Model	Exact Match	Field Accuracy	BLEU	ROUGE-L F1
LLaMA	0.0057	0.4710	0.1801	0.4035
Gemma	**0.0263**	0.5799	0.2731	0.4921
DeepSeek	**0.0263**	**0.6094**	**0.2891**	**0.5181**

The evaluation results show clear differences in model performance on the structured prediction task. While exact match scores are low across all models ($\leq 2.6\%$), this is expected given the strict nature of the metric—requiring every field in the structured output to be correct.

Looking beyond exact match, field-level accuracy and text generation metrics (BLEU and ROUGE-L F1) provide a more refined view:

- DeepSeek achieved the highest field accuracy (0.6094) and also led in both BLEU (0.2891) and ROUGE-L F1 (0.5181), suggesting it is most reliable at producing both correct field values and fluent text in **'response-to-sender'** and **'response-to-recipient'**.
- Gemma performed competitively, with a field accuracy of 0.5799 and slightly lower text quality metrics than DeepSeek, indicating a consistent but marginally less precise output.
- LLaMA, despite completing training, lagged behind both models in all metrics. Its low BLEU (0.1801) and ROUGE-L (0.4035) suggest challenges in generating fluent, correct text responses—likely impacted by the shorter training duration (2 epochs vs. 8).

Overall, DeepSeek appears to generalize best on this task, with Gemma close behind. LLaMA may require longer training or adjusted hyperparameters to match their performance.

5 Conclusion

This paper presents a practical framework for deploying LLMs in low-resource service delivery contexts, highlighting how carefully curated datasets, lightweight fine-tuning, and structured evaluation can yield useful automation tools. Rather than relying on general-purpose reasoning capabilities, we show that narrow, reliable models trained on real-world prompts can power transactional workflows such as reminders and scheduling.

We demonstrate that with minimal resources and careful annotation, it is possible to build and deploy efficient LLMs tailored for social service use cases.

Our results suggest that task-specific finetuned models offer a viable and accessible path for automating structured social interactions in emerging digital economies.

6 Limitations

Looking forward, there are several avenues for improvement. First, incorporating more African languages explicitly in the training (possibly via translation or community data efforts) would enhance the system's inclusivity.

References

Luca Beurer-Kellner, Marc Fischer, and Martin Vechev. Guiding llms the right way: Fast, non-invasive constrained generation, 2024. URL https://arxiv.org/abs/2403.06988.

Wei-Rui Chen, Ife Adebara, Khai Duy Doan, Qisheng Liao, and Muhammad Abdul-Mageed. Fumbling in babel: An investigation into chatgpt's language identification ability, 2024. URL https://arxiv.org/abs/2311.09696.

Saibo Geng, Hudson Cooper, Michał Moskal, Samuel Jenkins, Julian Berman, Nathan Ranchin, Robert West, Eric Horvitz, and Harsha Nori. Jsonschemabench: A rigorous benchmark of structured outputs for language models, 2025. URL https://arxiv.org/abs/2501.10868.

Michael A. Hedderich, Lukas Lange, Heike Adel, Jannik Strötgen, and Dietrich Klakow. A survey on recent approaches for natural language processing in low-resource scenarios, 2021. URL https://arxiv.org/abs/2010.12309.

Neal Jean, Marshall Burke, Michael Xie, W. Davis, David Lobell, and Stefano Ermon. Combining satellite imagery and machine learning to predict poverty. *Science*, 353:790–794, 08 2016. doi: 10.1126/science.aaf7894.

Yaxi Lu, Haolun Li, Xin Cong, Zhong Zhang, Yesai Wu, Yankai Lin, Zhiyuan Liu, Fangming Liu, and Maosong Sun. Learning to generate structured output with schema reinforcement learning, 2025. URL https://arxiv.org/abs/2502.18878.

Michal Lukasik, Harikrishna Narasimhan, Aditya Krishna Menon, Felix Yu, and Sanjiv Kumar. Regression-aware inference with llms, 2024. URL https://arxiv.org/abs/2403.04182.

N. Moorosi. Better data sets won't solve the problem — we need ai for africa to be developed in africa. *Nature*, 636(8042):276, 2024. doi: 10.1038/d41586-024-03988-w. URL https://doi.org/10.1038/d41586-024-03988-w.

J Rakesh, Riyaz Shaik, and Sai S V. Ai based virtual banking assistant. 11:274–284, 04 2025. doi: 10.5281/zenodo.15285327.

Connor Shorten, Charles Pierse, Thomas Benjamin Smith, Erika Cardenas, Akanksha Sharma, John Trengrove, and Bob van Luijt. Structuredrag: Json response formatting with large language models, 2024. URL https://arxiv.org/abs/2408.11061.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, Perry Payne, Martin Seneviratne, Paul Gamble, Chris Kelly, Nathaneal Scharli, Aakanksha Chowdhery, Philip Mansfield, Blaise Aguera y Arcas, Dale Webster, Greg S. Corrado, Yossi Matias, Katherine Chou, Juraj Gottweis, Nenad Tomasev, Yun Liu, Alvin Rajkomar, Joelle Barral, Christopher Semturs, Alan Karthikesalingam, and Vivek Natarajan. Large language models encode clinical knowledge, 2022. URL https://arxiv.org/abs/2212.13138.

Zhi Rui Tam, Cheng-Kuang Wu, Yi-Lin Tsai, Chieh-Yen Lin, Hung yi Lee, and Yun-Nung Chen. Let me speak freely? a study on the impact of format restrictions on performance of large language models, 2024. URL https://arxiv.org/abs/2408.02442.

Naizhuo Zhao, Katia Charland, Mabel Carabali, Elaine O. Nsoesie, Mathieu Maheu-Giroux, Erin Rees, Mengru Yuan, Cesar Garcia Balaguera, Gloria Jaramillo Ramirez, and Kate Zinszer. Machine learning and dengue forecasting: Comparing random forests and artificial neural networks for predicting dengue burden at national and sub-national scales in colombia. *PLOS Neglected Tropical Diseases*, 14(9):1–16, 09 2020. doi: 10.1371/journal.pntd.0008056. URL https://doi.org/10.1371/journal.pntd.0008056.