**Resources On Zambian Translation Tool**

A logo with a lock on it

AI-generated content may be incorrect.

**Abstract**

This tool is created in collaboration with the Zambian Cyber Security Initiative Foundation, 2025 under the project manager **John Tshinseki.** The main motive behind the development of this tool was to create a one-stop database for local Zambian languages. A tool that can help in translating local Zambian languages into English and vice-versa.

The tool is still in its development phase and supports local Zambian languages as listed below:

* Nyanja
* Bemba

Still under works to achieve translation on local dialects such as:

* Lozi
* Luvale
* Tonga
* Kaonde
* Lunda
* Ndebele
* Shona
* Zulu

This is a work in progress project with ability to release in near future.

**Research and Resources Used**

Over the course of this project, significant research has been conducted to develop a translator for lower-resourced African languages, with an initial focus on Nyanja (Chichewa).

Various open-source translation models, including OPUS-MT and M2M100, were assessed, but each demonstrated limited coverage or inconsistent performance for Nyanja. Ultimately, Meta’s **NLLB-200 Distilled 600M** model was selected for its improved support of African languages, being lightweight and the inclusion of the “nya\_Latn” language code.

To enhance translation consistency and quality, beam search and early stopping were introduced in the decoding phase. This refined approach helped reduce repetitive outputs and partial translations.

In parallel, the translator’s capabilities were extended to include **Bemba** (“bem\_Latn”), leveraging and training the same NLLB model on Bemba dataset. This expansion required identifying Bemba’s appropriate language code and conducting validation to ensure accurate results.

The translator is organized into two modules:

1. A **backend** (translator\_backend.py) is responsible for loading the NLLB model, applying beam search, and generating translations.
2. A **frontend** (translator\_frontend.py) built with **Streamlit** provides a user-friendly interface that allows individuals to select the desired language direction and quickly obtain a translated text.

By combining these Natural Language Processing techniques with straightforward user interaction, this project tries to demonstrate a practical application of local Zambian language translation technology.

Following, the further links to technologies used up until now:

* Documentation on Opus-Mt:

<https://github.com/Helsinki-NLP/Opus-MT>

<https://huggingface.co/Helsinki-NLP/opus-mt-en-mt>

* Documentation on NLLB:

<https://huggingface.co/docs/transformers/en/model_doc/nllb>

* Documentation on NLLB-200-distilled-600M:

<https://huggingface.co/facebook/nllb-200-distilled-600M>

* Documentation on Streamlit:

<https://docs.streamlit.io/>

* Documentation on Bemba dataset used for training:

<https://github.com/csikasote/bigc>