Abstract

Yoruba is a West African language spoken by approximately 47 million people worldwide. Though its roots lie with the Yoruba people primarily residing in Nigeria, the language extends dialectically across several countries, including Benin, Togo, Ghana, Sierra Leone, Brazil, and The Gambia. Despite this wide geographic and cultural reach, there exists a surprising lack of comprehensive online tools for learning, teaching, or documenting the Yoruba language.

A basic Google search for "Yoruba Dictionary" yields a handful of results—such as YorubaDictionary.com, Glosbe, and Lexilogos—yet each of these falls short in key areas that The Yoruba Dictionary aims to address. Many existing platforms lack essential search capabilities, rely heavily on Google Translate without independent linguistic verification, or do not support advanced features such as voice search through automatic speech recognition (ASR) or text-to-speech (TTS) functionality in Yoruba. This technological gap is likely due to the limited size of existing Yoruba language corpora, which has impeded the development of robust ASR and text-to-speech models tailored to the language.

I developed *The Yoruba Dictionary*, a mobile-accessible and easily maintainable web application to support the learning of Yoruba using the formal vocabulary taught in Yale's Yoruba language curriculum. It is the culmination of a semester-long project combining good user interface design and server management with ASR and text-to-speech techniques to showcase the Yoruba language. It includes a rudimentary text-to-speech voice synthesis model built by the concatenation of Yoruba phonemes to pronounce words. While the ASR component for Yoruba voice search is not yet fully implemented, the platform is under active development and aims to evolve into a comprehensive resource for students and Yoruba language learners alike.

Introduction

Despite significant advancements in natural language processing (NLP) over the past couple of decades, Yoruba remains largely underrepresented in this technological space. This is primarily due to market non-interest—research and development tend to prioritize languages with larger user bases or seemingly greater commercial potential, leaving many African languages, including Yoruba, without sufficient investment. Another major barrier is the lack of a comprehensive and digitized linguistic corpus for Yoruba, which is essential for training effective NLP models. Corpus signifies the range of annotated texts and audio data that allow machine learning systems to understand and process language accurately. Yoruba, however, suffers from limited standardized digital resources, compounded by complex linguistic features like tonal variation, which are often ignored in existing datasets. As a result, Yoruba is classified as a *low-resource language*, a term used in NLP to describe languages that lack the large-scale, high-quality data necessary for building state-of-the-art language models. These factors may have hampered the development of tools like speech recognition, machine translation, and text-to-speech systems for Yoruba. This project, *The Yoruba Dictionary*, set out to change that by building a scalable, intuitive, and linguistically accurate web application aimed at learners, educators, and researchers of the Yoruba language.

Related Work

A few Yoruba dictionary tools and translation platforms exist online, offering varying levels of functionality, depth, and linguistic accuracy. **Glosbe** is one of the most prominent resources, functioning as a crowdsourced multilingual dictionary that includes some Yoruba. It provides word-level translations, example sentences, and community-contributed definitions. However, its reliability can vary, as entries are user-generated and lack linguistic review. Additionally, tonal markings are inconsistently applied, which can be problematic for learners and speakers who rely on accurate tone for meaning. It still is, however, one of the better resources that exist as an online Yoruba dictionary.

Google Translate, though widely accessible, provides only basic Yoruba translation capabilities. Its word-for-word translations frequently ignore grammatical structure and context, leading to significant inaccuracies. Even for each word, Google Translate often fails to give a correct translation. For instance, tonal variations in Yoruba words, which can completely change meaning, are not accounted for in Google's system. As a result, it may be helpful for very common words but is generally unreliable for learning Yoruba vocabulary or phrases used in context.

Other platforms, like **Yorubadictionary.com**, have very basic search algorithms developed that don't work in the majority of search contexts, or like **Lexilogos** serve as an aggregated interface to the aforementioned sites. For these all of the tools such a hub links to are external and sometimes outdated.

In the automatic speech recognition scene, foundational work in low-resource language settings have existed and continue to be developed. Some amongst them are OpenAI's Whisper and CMU Sphinx, open-source ASR models trained on labeled multilingual audio data collected from the web. Whisper demonstrates impressive generalization across many languages, but its performance on low-resource and tonal languages such as Yoruba is limited, largely due to data scarcity and insufficient modeling of tonal information. My project did not have the time to develop an ASR model, but in the future, I believe meta-learning multilingual models like Whisper would be worth continuing to look into in order to create a foundational ASR model in Yoruba.

Additionally, much discourse on other techniques for text-to-speech or automatic speech recognition has been done, and specifically a couple papers are worth mentioning. [Gauthier et al., 2016] discusses automatic speech recognition for Hausa and Wolof taking into account the languages unique characteristics in vowel length contrast. Other papers, like [Ananthi, S., Dhanalakshmi, P., 2015] and [A. Pradhan et al., 2013] compare statistical techniques to synthetic speech synthesis using syllabic units, which is a similar approach to the one I take in the Yoruba Dictionary.

User Interface Design & System Architecture

Focusing first on the architecture of the website, *The Yoruba Dictionary* is built as a full-stack web application with a React frontend served through an nginx reverse proxy on an Ubuntu Linux server, provisioned via Yale SpinUp, hence its domain name (.yale.edu). Real-time interaction is resolved between React's responsive routing and design and API calls to a separately developed backend.

The architecture of my data backend changed repeatedly over the course of development of the dictionary. Initially, the backend served dictionary entries directly from the locally stored Excel spreadsheets, which was the Yoruba glossary, using a lightweight Flask server. However, due to latency and reliability concerns, as calls to our database would either have to return the entire database or parse and search through the database on every request (and until fixed, required a spin-up time before service), I migrated the data layer to Google's Firebase Firestore. Firestore offered persistent, indexed storage with low-latency reads and write operations, as well as automatic scalability for adding more words and phrases to our glossary. In addition, local session storage on the client side was introduced to cache frequent lookups and reduce redundant database access. The original Flask backend, as well as the original excel spreadsheets, were removed from the project.

Upon the development of my text-to-speech model, another backend had to be introduced to accept requests for text input and output a synthesized audio file to be played by the front-end. This initially started as a Node Express.js backend, but was migrated to a Flask Python server due to the comparative benefit I found utilizing Python's existing audio manipulation libraries. A detailed breakdown of my TTS model will be provided later.

The search functionality uses a hybrid scoring algorithm based off of Levenshtein distance for string matching, and it combines proportional string similarity with weighted metadata matching. User search queries are not only evaluated against headwords, but also alternative translations, topics, example usage, and parts of speech, with decreasing weight multipliers assigned to these metadata fields. The system supports searches in both English and Yoruba, allowing the user to find both Yoruba translations for English words and vice versa.

To accommodate Yoruba's tonal system—where identical spellings may carry different meanings based solely on pitch— as well as English's lack of such thereof, the interface includes a custom tone keyboard for diacritic input (e.g., à, á, ẹ̀, ọ́, ṣ) and relaxed character matching for untoned queries. These design considerations ensure the tool remains accessible to novice users unfamiliar with tone placement, while still maintaining linguistic fidelity for advanced learners and researchers.

NLP Features Implementation

The text-to-speech component of *The Yoruba Dictionary* began with an attempt to integrate Google Cloud's Text-to-Speech (TTS) API to provide pronunciation for Yoruba words. However, this approach proved unviable: Yoruba is not supported as a language option within the API, and substituting with somewhat phonetically similar languages like Spanish resulted in inaccurate pronunciations. These Spanish models failed to produce key Yoruba phonemes, such as “gb,” “kp,” and various tonal vowels, like the long a sound, leading to poor results. Attempts to “respell” Yoruba words using their phonetic counterparts in different languages also didn’t work at all, especially since even within some languages, the same groups of characters would produce inconsistent pronunciations depending on the surrounding characters!

To address this, I changed my primary focus from developing an ASR model to a TTS one, realizing text-to-speech was not nearly as trivial as I thought it would have been. The result was my custom syllable-based TTS system. Upon receiving a text query (the Yoruba word that’s supposed to be heard aloud), my system algorithmically parses the text into its component syllables—phonemes that map directly to how the language is spoken and written. Using a Flask application, I recorded my own voice pronouncing a range of monosyllabic Yoruba phonemes, which were then stored and indexed in Firestore. When a word is queried, the backend synthesizes a new syllable recording by mixing the given data based on the phoneme requested and concatenates them into a single audio output. This approach seemed particularly effective because Yoruba syllables follow a consistent and regular phonetic structure, enabling synthesis from discrete phoneme recordings without a huge amount of data, but it also has its flaws, with awkward pauses within words, sometimes faulty phoneme synthesis, and with more data, may not necessarily perform better. It seems to work alright for the current use cases, however.

Looking forward, the next phase involves developing an automatic speech recognition (ASR) system to support Yoruba voice search. I plan to apply a similar syllable-based methodology to train or fine-tune a model—potentially using Whisper or another open-source speech recognition framework—on annotated Yoruba syllable data. Hopefully, this would allow for accurate, tonal-aware transcription of user speech into Yoruba text.

Evaluations

The majority of current evaluations on the Yoruba Dictionary site are testers and my advisor, Oluseye Adesola, Senior Lector II in Yoruba & African Studies at Yale. Their reception of the website has been generally positive, and remarks on the site are that it is very useful for recalling forgotten phrases or learning new ones. Adesola specifically mentioned that the text-to-speech model currently available in *The Yoruba Dictionary* (at https://yorubadictionary.yale.edu) "shows good progress, but still needs work on its flow and tones”.

Future Work

Future work on *The Yoruba Dictionary* includes several key areas of expansion. First, implementing voice input search through automatic speech recognition (ASR) would allow users to speak queries naturally, leveraging tonal information inherent in speech. This could be achieved by training or fine-tuning models like Whisper on Yoruba syllable-level data. Second, the platform could support interactive learning modules—such as quizzes, audio exercises, and video lessons—to enhance engagement and reinforce vocabulary and grammar. Continuing to expand the dictionary corpus with additional texts, proverbs, and oral literature would improve linguistic coverage and cultural depth. Additionally, developing tools for community contributions—allowing speakers, educators, and students to suggest edits, add examples, or contribute audio—would help keep the resource up-to-date and collaborative. These developments would transform *The Yoruba Dictionary* into a dynamic, multi-functional platform for language preservation and education.

Conclusion

In conclusion, *The Yoruba Dictionary* represents a meaningful step forward in creating accessible, digital infrastructure for a historically under-resourced language. Over the course of the project, the platform evolved into a robust, searchable, and mobile-accessible application with a custom search engine, a structured Firestore database, and a rudimentary syllable-based text-to-speech model. Beyond its technical achievements, this project underscores the importance of digital tools in preserving linguistic heritage. By making Yoruba more accessible to learners, educators, and researchers, this platform helps counter the gradual erosion of vocabulary and fluency among speakers and offers a scalable model for supporting other low-resource languages through technology.

References

[1] A. Pradhan, A. S. Shanmugam, A. Prakash, K. Veezhinathan, and H. Murthy. 2013. A syllable based statistical text to speech system. In *Proceedings of the 21st European Signal Processing Conference (EUSIPCO 2013)*, Marrakech, Morocco, 1–5.

[2] E. Gauthier, L. Besacier, and S. Voisin. 2016. Automatic speech recognition for African languages with vowel length contrast. *Procedia Computer Science* 81 (2016), 136–143. DOI:https://doi.org/10.1016/j.procs.2016.04.041

[3] S. Ananthi and P. Dhanalakshmi. 2015. Syllable based concatenative synthesis for text to speech conversion. In *Computational Intelligence in Data Mining – Volume 3*, L. Jain, H. Behera, J. Mandal, and D. Mohapatra (Eds.). Smart Innovation, Systems and Technologies, Vol. 33. Springer, New Delhi. DOI:https://doi.org/10.1007/978-81-322-2202-6_6