The Yoruba Dictionary: Crafting A Online Platform for A Low-Resource Language

Muyi Aghedo, Computer Science, Oluseye Adesola, Yoruba & African Studies, Yale University

Overview of The Yoruba Dictionary

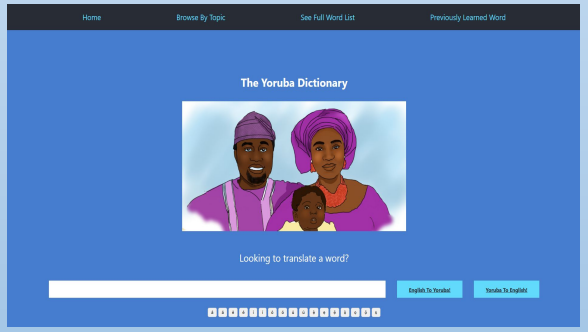
🧠 **Why It Matters**

- Yoruba, and African languages in general, should have digital tools. The importance of preserving the language, by adding resources for its accessibility and education cannot be overemphasized.
- Yoruba itself has around 47 million speakers globally.

📈 **Future Roadmap**

- Automatic speech recognition and voice search
- Community contributions (add words/ more audio samples)
- Interactive quizzes, language games, video modules
- An expanded corpus with proverbs, idioms, and stories

User Interface & Design



Modern, Mobile-Friendly Web App
Built with a responsive React frontend, optimized for accessibility on phones, tablets, and desktops.

Bi-Directional Search Engine
Users can search in either Yoruba or English, with matching across headwords, translations, parts of speech, and example usage.

Smart Search Matching
Utilizes a hybridized Levenshtein distance algorithm for typo and tone-tolerant searching

Tone-Friendly Input System
Custom Yoruba tone keyboard for accented characters (à, á, ẹ, ṣ, etc.), plus relaxed matching for users unfamiliar with diacritics.

Efficient Data Accesses & Scalable Data Layer
Firebase Firestore backend ensures fast, indexed lookups with client-side caching to minimize latency.

Text-to-Speech Functionality (TTS)
Unique syllable-based TTS system using pre-recorded phoneme clips to vocalize Yoruba words—custom-developed due to lack of Yoruba support in common APIs.

https://yorubadictionary.yale.edu

Custom-Built Text-To-Speech System

Syllable-Based Concatenative Synthesis
Words are broken down into monosyllabic phonemes, which are then stitched together to form speech.

Recorded with Native Pronunciation
The developer recorded their own voice to ensure accurate tonal and phonemic representation of Yoruba syllables.

Dynamic Audio Generation
On receiving a word, the backend retrieves and concatenates relevant audio clips to produce a complete pronunciation.

Challenges & Improvements Ongoing
Current version may include slight pauses or artifacts, but it's a strong foundation for future refinement.

```
for word in example_words:
  print(breakdownWord(word))
```

```
['ì', 'wá', 'ṣẹ́']
['a', 'kọ́', 'ṣẹ́', 'mọ́', 'ṣẹ́']
['ẹ̀', 'san']
['à', 'bí', 'gbẹ̀', 'yìn']
['a', 'mò', 'fin']
['kọ́', 'bọ́', 'ọ̀', 'dù']
en
i
['be', 'e', 'ni']
['e', 'gun', 'gun']
['ke', 'hin', 'de']
['mo', 'ma', 'a', 'n', 'jo', 'ko', 'o.']
n
i
['mo', 'ma', 'a', '', 'ni', 'wa', 'se']
```

Hybridized Similarity Search Algorithm

Candidate Vocabulary Word	Distance to Query: "ọrẹ"	Search Score
ọ̀rẹ́	0 + 2λ	100% - 2kλ
ó rẹ̀	1 + 2λ	75% - 2kλ
rorẹ́	1 + 2λ	75% - 2kλ
ọ̀ṣẹ̀	1	66%