Text-to-Speech System for Hindi Language: A Syllable-based Approach

Priyanshu Jha(231110039) Neelu Lalchandani(231110031)
Tanmay Dubey(231110052)

23/04/2024

Abstract

This project presents a Text-to-Speech (TTS) system for the Hindi language, leveraging a syllable-based approach. The system takes Hindi text as input and generates an audio output containing the spoken form of the text. The project is implemented using Python programming language and utilizes various components to handle different stages of the TTS pipeline. The system maps input text to a sequence of phonemes, which are then mapped to pre-recorded audio files corresponding to individual syllables. These audio files are combined and processed to generate the final speech output. The audio file database is created by us with the help of Google TTS where we have recorded near about 450 audio files in wav format. The project demonstrates the efficacy of the syllable-based approach in handling the complexities of the Hindi language and its writing system.

1 Introduction

Text-to-Speech (TTS) technology has garnered significant attention in recent years due to its numerous applications in various domains, including assistive technologies for visually impaired individuals, multimedia applications, language learning tools, and virtual assistants. TTS systems enable computers to generate human-like speech output from written text, facilitating seamless communication and enhancing accessibility.While TTS systems for languages with relatively simple writing systems, such as English, have achieved remarkable success, developing TTS solutions for languages with complex writing systems, such as Hindi, presents unique challenges. The Hindi language, written in the Devanagari script, incorporates a vast number of consonant-vowel combinations, diacritical marks (matras), and intricate phonetic structures, making it difficult to accurately map text to speech using traditional phoneme-based approaches.

To address these challenges, this project proposes a novel syllable-based approach for TTS in the Hindi language. Unlike phoneme-based methods, which can struggle with the complexities of the Devanagari script, the proposed system breaks down input text into syllables and maps them to pre-recorded audio files representing those syllables. By combining these audio files, the system generates natural-sounding speech output while accounting for the nuances of the Hindi language. The project employs a modular architecture, consisting of several components that handle various tasks in the TTS pipeline, including text preprocessing, tokenization, syllabification, phoneme conversion, and audio processing. The system follows a rule-based approach, incorporating linguistic knowledge of the Hindi language and its writing system to accurately handle the complexities of the Devanagari script.

Moreover, the project leverages a dataset of around 450 recorded audio files whose recording we have done using google TTS, each representing a unique syllable in the Hindi language. By addressing the challenges posed by the Hindi language's complex writing system and leveraging a syllable-based approach, this project aims to contribute to the development of assistive technologies, multimedia applications, and language learning tools for the Hindi-speaking community, ultimately promoting accessibility and inclusivity.

2 Components

1. **Tokenizer**: The tokenizer is responsible for preprocessing the input text and breaking it down into individual words or tokens.

2. **Syllabifier**: The syllabifier component plays a crucial role in breaking down each word into its constituent syllables.

3. **Word-to-Phoneme Converter**: The word-to-phoneme converter component maps the list of syllables for each word to their corresponding phoneme representations.

4. **Phoneme-to-Sound Mapper**: The phoneme-to-sound mapper component is responsible for mapping each phoneme in the sequence to a corresponding pre-recorded audio file (WAV file).

5. **Audio Combiner**: The audio combiner component combines the individual audio files corresponding to the phoneme sequence, generating the final speech output.

6. **Char-to-Phoneme ID Mapping**: This module contains mappings between characters or syllables in the Devanagari script and their corresponding phoneme IDs or audio file names.

7. **Phonemes Module**: The phonemes module serves as a central repository for phoneme definitions and constants used throughout the TTS system.

3 Dataset

The project utilizes a dataset of around 450 audio files, each representing a unique syllable in the Hindi language. We have created these audios using Google TTS library. It includes 'Swaras', 'Vyanjanas' and the 'Barakhadi' of them. It also includes most frequent bigram syllables. The availability of a comprehensive dataset of syllable recordings is crucial for the successful implementation of the syllable-based approach.

4 Methodology and Proposed Approach

The TTS system for the Hindi language follows a modular and rule-based approach, leveraging the specific characteristics of the Hindi language and its writing system. The system is designed to handle the complexities of the Devanagari script, including its vowels, consonants, and diacritical marks (matras).

The overall methodology can be divided into the following key stages:

1. Text Preprocessing and Tokenization
2. Syllabification
3. Word-to-Phoneme Conversion
4. Phoneme-to-Sound Mapping
5. Audio Combination and Processing

By following this methodology and leveraging the specific characteristics of the Hindi language and its writing system, the TTS system is able to generate high-quality speech output from input text. The rule-based approach and the use of recorded audio files contribute to the system's effectiveness in handling the complexities of the Devanagari script and producing natural-sounding speech output.

5 Conclusion

The Text-to-Speech system for the Hindi language, developed using a syllable-based approach, demonstrates the effectiveness of this technique in handling the complexities of the Hindi language and its writing system. By breaking down input text into syllables and mapping them to pre-recorded audio files, the system generates high-quality speech output.

The modular design of the project allows for easy maintenance and potential future enhancements. The system can be further improved by expanding the dataset of pre-recorded audio files, incorporating more advanced techniques for syllabification and phoneme conversion, and exploring alternative approaches to audio combination and processing.

Overall, this project contributes to the development of assistive technologies and multimedia applications for the Hindi language, enabling better accessibility and enhanced user experiences.

6 Limitations

1. Our model is not designed to handle numbers and special symbols currently.
2. Our model currently is currently capable only to handle top 60 bigrams and no trigram or n-gram.
3. Our dataset currently has nearabout 450 audiofiles recorded which can be increased to obtain better fluency in speech generated.