An HMM-Based System for Speech Recognition and Pronunciation Generation

PRESENTED BY:

AKSHARA SHARMA

(A023167022060)

7 CSE – DS 1Y

Introduction and Project Objectives

Problem Domain

- The modeling of speech signals presents a significant computational challenge due to the inherent variability in human elocution. This project addresses the need for interpretable and data-efficient models for fundamental speech processing tasks.

Project Objectives

- The primary objective was to design, implement, and evaluate a unified system capable of both analyzing and synthesizing speech, based on the principles of Hidden Markov Models.
- **Speech Recognition:** To develop an isolated word speech recognizer with high classification accuracy, demonstrating the efficacy of HMMs in a constrained-vocabulary task.
- **Pronunciation Generation:** To construct a synthesis module capable of generating accurate phonetic transcriptions and intelligible audio for an arbitrary vocabulary.
- This work focuses on HMMs to explore their strengths in transparency and performance within well-defined problem spaces.

Hidden Markov Models (HMMs)

A **Hidden Markov Model** is a statistical model that represents a system as a Markov process with unobserved (hidden) states. HMMs are exceptionally well-suited for modeling time-series data, such as speech.

Key Components of an HMM:

- **Hidden States (S):** A set of unobservable states. In speech, these correspond to abstract phonetic units.
- **Observations (O):** A sequence of observable outputs generated by the states. In speech, these are the acoustic feature vectors (MFCCs).
- **Transition Probabilities (A):** The probability of moving from one hidden state to another.
- **Emission Probabilities (B):** The probability of emitting a specific observation from a given hidden state.
- By training an HMM, we learn the optimal transition and emission probabilities that best explain the observed data.

Acoustic Feature Extraction: MFCCs

Directly processing raw audio waveforms is computationally inefficient. **Mel-Frequency Cepstral Coefficients (MFCCs)** are the canonical features used in speech recognition for their robustness and perceptual relevance.

Rationale for MFCCs:

- **Bio-Mimetic:** The Mel-frequency scale approximates the non-linear frequency response of the human cochlea, providing greater resolution at lower frequencies, which are critical for speech intelligibility.
- **Dimensionality Reduction:** The process transforms high-dimensional, correlated audio signals into a low-dimensional, decorrelated set of feature vectors, which serve as a compact acoustic signature.
- The extraction process involves framing, Fourier analysis, Mel-filtering, and a discrete cosine transform to produce the final coefficient vectors.

Speech Recognition Pipeline

- The recognition process classifies an unknown audio signal by determining which pre-trained HMM most likely generated it.
- **Audio Input:** An unclassified .wav file is ingested.
- **MFCC Extraction:** The signal is converted into a sequence of MFCC feature vectors.
- **Likelihood Calculation:** This sequence is evaluated against every trained word-specific HMM (e.g., HMM-apple, HMM-banana) using the Viterbi or Forward algorithm to calculate the log-likelihood score.
- **Maximum Likelihood Selection:** The HMM that yields the highest log-likelihood score is selected as the most probable source model.
- **Classification Output:** The label associated with the winning HMM is returned as the recognized word.

Synthesizing New Words from Learned Phonemes

This pipeline leverages HMMs trained on individual phonemes from the **fruit dataset** to construct the audio for any new word.

Workflow:

- **Build Phoneme Library:** The audio from the fruit dataset is first used to train a reusable HMM for each unique phoneme (/æ/, /p/, /m/, etc.). This library becomes the system's acoustic foundation.
- **Deconstruct New Word:** An input word (e.g., "computer") is converted into its target sequence of phonemes using a G2P engine.
- **Reconstruct from Library:** The system retrieves the corresponding HMM from the library for each phoneme in the sequence, generates its unique sound, and stitches the segments together to create the final audio output.

THANKYOU !