Speech Recognition with Hidden Markov Models (HMMs)

Group Project Report

May 28, 2025

Introduction

This report presents our work on building a speech recognition system using **Hidden Markov Models (HMMs)**. Due to a widespread blackout in Spain during our academic term, we lost a full week of classes. Unfortunately, this affected our learning continuity, especially on advanced topics such as HMMs which were not fully covered before this assignment was given.

Despite these difficulties, we approached this project with the aim to learn practically. We relied on a Jupyter notebook shared publicly, which provided a technical walkthrough of HMM-based speech recognition. Our goal here is to explain what we understood and implemented in the most clear and human way possible.

What is a Hidden Markov Model?

A **Hidden Markov Model** (HMM) is a tool used to model systems that change over time in a probabilistic way. It's called "hidden" because we can't directly observe the states of the system — we only see outcomes (called *observations*) that are produced by those hidden states.

In speech recognition, the hidden states might represent different phonemes or sound units of speech, while the observations are measurements of the audio signal — for example, a type of feature called MFCC (Mel-Frequency Cepstral Coefficients).

An HMM is defined by:

- A set of **states** (e.g., different sounds or phonemes).
- A **transition probability** matrix (the probability of moving from one state to another).
- An **emission probability** distribution (the probability of observing a feature given the hidden state).
- An **initial state distribution** (which state we start in).

The idea is to learn the parameters of this model from labeled audio data. Once trained, given a new sequence of audio features, we can use the model to determine which word (or phoneme sequence) was likely spoken.

Step-by-Step Pipeline

The notebook we followed implements an end-to-end system for speech recognition using HMMs. Below, we summarize and expand each step we took.

1. Audio Input and Visualization

We began by reading '.wav' files using the `scipy.io.wavfile` module. This gives us both:

- The **sampling frequency** (e.g., 16,000 Hz).
- The raw **audio waveform**, which is just a sequence of numbers representing sound pressure.

We plotted the audio waveform to visualize how sound varies over time. This helps us understand the data before processing it.

2. Preprocessing and Normalization

Speech signals can vary in loudness and clarity depending on how they were recorded. To make all samples more consistent, we:

- Normalized the audio so its amplitude ranges between -1 and 1.
- Applied RMS (Root Mean Square) scaling to ensure uniform loudness.

We also handled stereo audio by selecting one channel or averaging both to create a mono signal.

3. Feature Extraction – MFCCs

Rather than using the raw waveform (which has thousands of values), we extracted features called **MFCCs** (Mel-Frequency Cepstral Coefficients).

MFCCs are compact numerical representations of the speech signal that:

- Emphasize perceptually important information.
- Reduce the dimensionality of the data.
- Are commonly used in speech and audio classification tasks.

We also computed **delta** and **delta-delta** features (which represent the speed and acceleration of the MFCCs over time), making our feature vectors more descriptive.

4. Data Augmentation

To improve model robustness and reduce overfitting, we artificially created more audio data using:

- **Pitch shifting** (making audio sound higher or lower).
- **Speed changes** (making speech faster or slower).
- **Time stretching** (changing duration without altering pitch).

This helped the model generalize better to different voices and recording conditions.

5. HMM Training per Word

We trained a separate HMM for each word in the dataset. That means:

- All audio examples for the word "yes" were used to train one HMM.
- All examples for "no" trained a different HMM.

Each HMM learned the statistical structure (patterns) of the audio features representing its corresponding word.

6. Classification via Scoring

Once all HMMs were trained, we used them to classify new audio samples:

- For each sample, we calculated the **log-likelihood score** under every HMM.
- The model that gave the highest score was assumed to be the predicted class.

To improve this further, these scores were used as features for a simple **logistic regression classifier**, providing a final decision.

Results and Evaluation

We evaluated the system using:

- **Confusion matrices** to see which words were confused.
- **Accuracy scores** to get a sense of overall performance.

Despite the complexity of speech recognition and our short timeline, the model showed encouraging results and worked reliably for basic commands.

Reflections and Conclusion

This project was a valuable learning experience. Working under the pressure of a lost week due to a national power blackout — with reduced access to classes, guidance, and time — was extremely challenging.

We did not fully understand HMMs at the start. The theory is complex and we lacked time to deeply study the math behind it. However, by following a structured and practical approach, we were able to build a working system and, in the process, gained a meaningful understanding of how HMMs apply to real-world audio.

Things We Learned

- The importance of preprocessing and normalization in audio tasks.
- How MFCCs transform raw sound into meaningful features.
- The intuition of HMMs — modeling sequences of data with state transitions.
- That data augmentation is key for performance, even in simple models.

Final Thoughts

This report does not present a perfect or deeply academic solution, but rather a sincere and resilient attempt by students facing real obstacles. We hope our work reflects a spirit of effort, learning, and adaptation.

Acknowledgment: We thank the creator of the original notebook (Ashiq Nazir Bhat), whose clear and well-commented code was essential to our progress. We also thank our instructors for their flexibility during a difficult time.