Audio Command Classification for a Voice-Activated Cooking Assistant

Rim Barbar

April 26, 2025

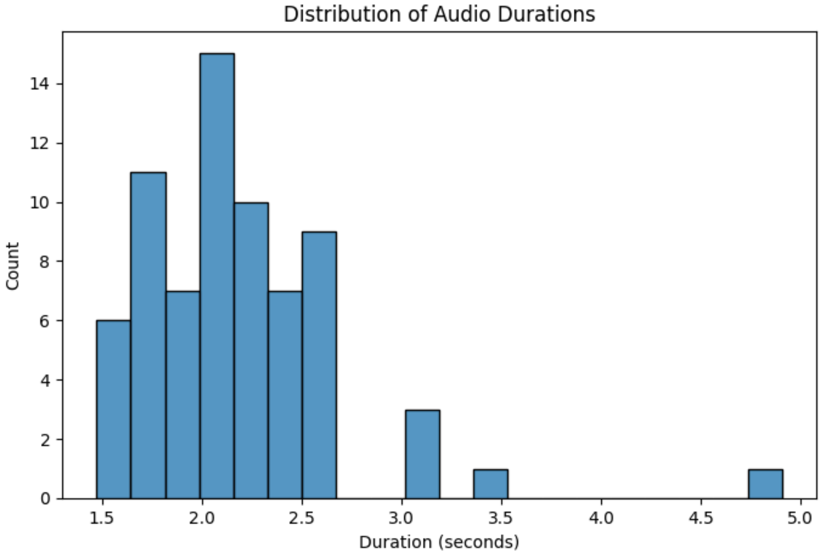
Lighthouse Labs Data Science Final Project

1. Introduction

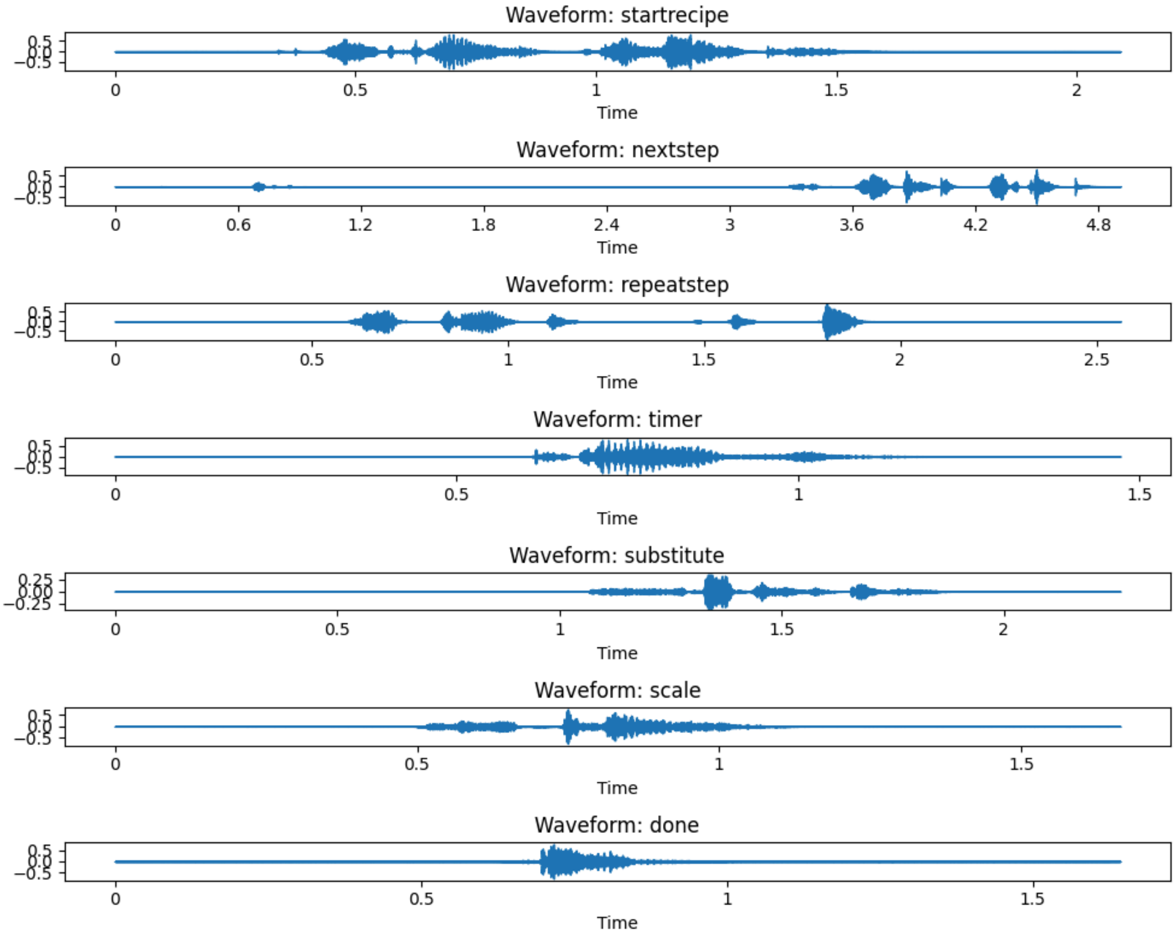
- **Purpose**: Develop a voice-activated system to classify 7 cooking commands (startrecipe, nextstep, repeatstep, timer, substitute, scale, done) to assist users in hands-free cooking.
- **Motivation**: Voice interfaces improve accessibility and convenience in the kitchen, but small datasets and audio variability pose challenges.
- **Objective**: Fine-tune a facebook/wav2vec2-base model to achieve high accuracy on a 70-sample dataset using 5-fold cross-validation.
- **Scope**: Focus on preprocessing, EDA, model training, and performance analysis in Google Colab.

2. Dataset

- **Source**: 70 audio clips (.m4a converted to WAV) from /content/LL_final_project/, organized into /content/data/<command>/.
- **Characteristics** (from audio_metadata.csv):
  - 10 clips per class (balanced: 10 each for startrecipe, nextstep, etc.).
  - Duration: 1.2–4.5 seconds (mean ~2.5s).
  - Sample rate: 16kHz.
- **EDA Visualizations**:
  - **Duration Distribution** (duration_distribution.png): Most clips are 1–3 seconds, with a slight left skew (Figure 1).



- ○ **Sample Waveforms** (sample_waveforms.png): Each class has distinct patterns, but overlap (e.g., scale vs. done) may cause misclassifications (Figure 2).



- **Challenges**: Small dataset (70 samples, 56 train/14 validation per fold) limits generalization. Potential class overlap in audio features.

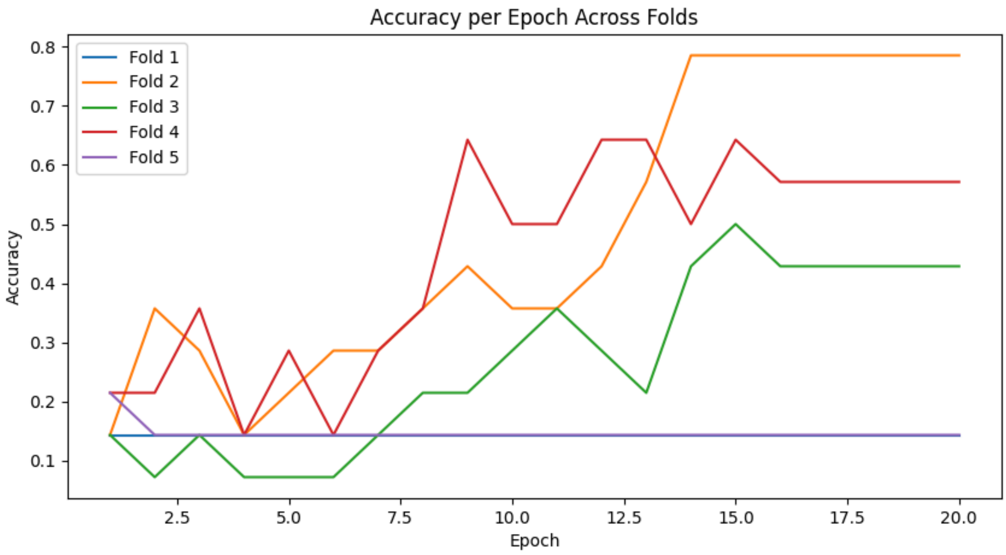
3. Methodology

- **Preprocessing**:
  - ○ Converted .m4a to WAV using pydub.
  - ○ Extracted features with Wav2Vec2Processor (16kHz, padded inputs).
- **EDA**:
  - ○ Analyzed metadata (audio_metadata.csv): Duration, sample rate, class balance.
  - ○ Visualized durations and waveforms to understand data variability.
- **Model**:
  - ○ facebook/wav2vec2-base fine-tuned for sequence classification (7 classes).
  - ○ Used StratifiedKFold (5 folds) to ensure balanced splits.

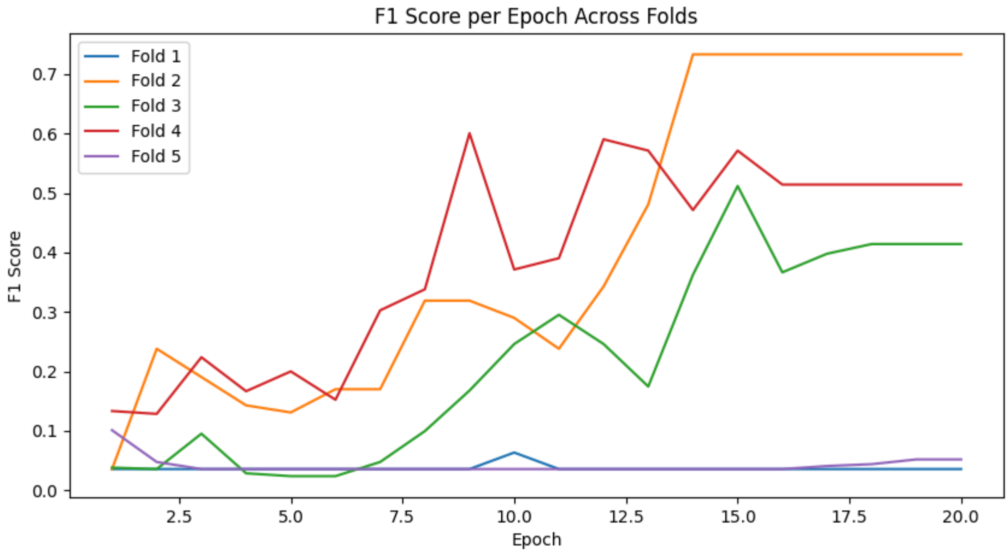
- **Training**:
  - 20 epochs, learning rate 5e-5, batch size 4, cosine scheduler.
  - Applied label smoothing (0.1), weight decay (0.01), gradient clipping (1.0).
  - Metrics: Accuracy, F1, precision, recall (via compute_metrics).
- **Evaluation**:
  - Plotted accuracy/F1 per epoch and final fold accuracies.
  - Noted precision warnings (some classes not predicted, especially in Folds 1 and 5).

4. Results

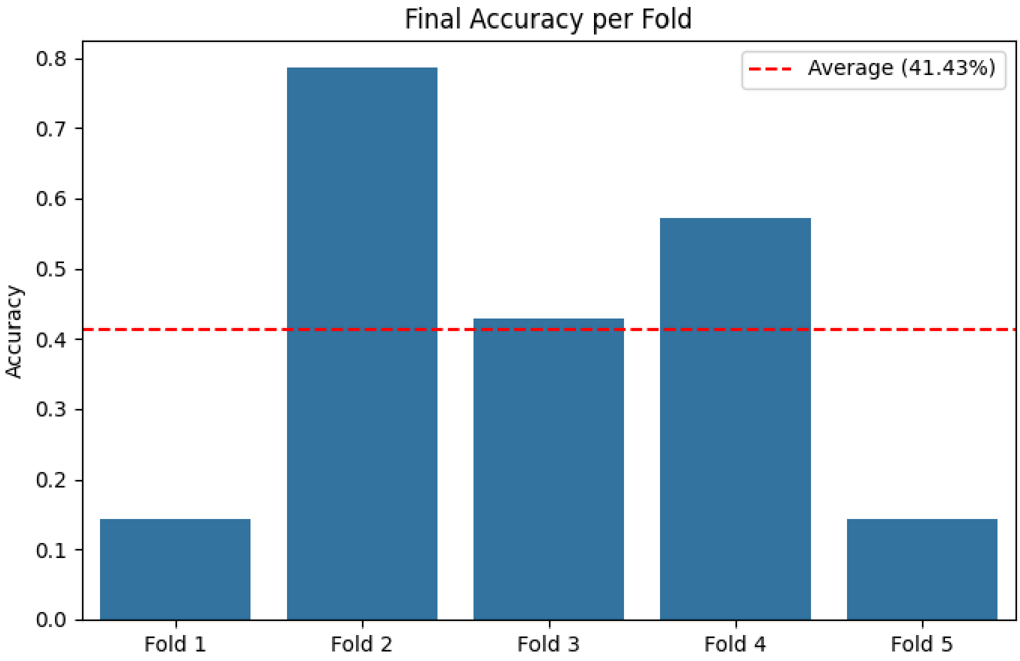
- **Performance** (from training logs):
  - **Average Accuracy**: 41.79% across 5 folds.
  - **Fold Breakdown**:
    - Fold 1: 14.29% (no learning, flat metrics).
    - Fold 2: 78.57% (best, F1 = 0.7333).
    - Fold 3: 42.86% (moderate improvement).
    - Fold 4: 57.14% (peaks at 64.29%).
    - Fold 5: 14.29% (no learning).
- **Visualizations**:
  - **Accuracy per Epoch** (accuracy_per_fold.png): Fold 2 peaks at 78.57%, others stagnate (Figure 3).



  - **F1 per Epoch** (f1_per_fold.png): Similar trends, Fold 2 at 0.7333 (Figure 4).



- **Final Accuracy** (final_accuracy_bar.png): High variance (14.29%–78.57%), mean 41.79% (Figure 5).



- **Insights**:
  - Fold 2's success shows model potential, but small dataset and poor validation splits (Folds 1, 5) limit performance.
  - Precision warnings suggest some classes (e.g., timer, substitute) are rarely predicted, reducing F1 scores.

5. Discussion

- **Successes**:
  - Demonstrated feasibility of classifying cooking commands with wav2vec2.
  - Fold 2's 78.57% accuracy suggests the model can learn with sufficient data.
  - EDA provided clear insights into data limitations (small size, class overlap).
- **Limitations**:
  - Small dataset (70 samples) leads to high variance (14.29%–78.57%).
  - Folds 1 and 5 failed to learn, possibly due to unbalanced validation splits despite StratifiedKFold.
  - Precision issues indicate model struggles with certain classes.
- **Challenges**:
  - Limited time prevented deeper analysis (e.g., confusion matrix).
- **Future Work**:
  - Collect more data (>100 samples per class).
  - Apply stronger augmentations (e.g., noise, pitch shift via audiomentations).
  - Test lower learning rates (e.g., 1e-5) or more epochs (25).
  - Deploy as a Streamlit app for real-time command recognition.

6. Conclusion

- The project achieved a 41.79% average accuracy, with Fold 2's 78.57% showing promise for voice-activated cooking assistants.
- Small dataset size and class overlap were major limitations, but EDA and visualizations provided valuable insights.
- Future improvements in data collection and model tuning could enable practical deployment.

References

- Hugging Face Transformers: facebook/wav2vec2-base.
- Libraries: transformers, datasets, pydub, audiomentations, seaborn, matplotlib, scikit-learn.