Chapter 4: Implementation

4.1 Development Environment

Hardware

Software Stack

Component   Version   Purpose

Virtual environment (venv) used for dependency isolation.

4.2 Data Processing Pipeline

Corpus Organisation

`

Filename format: nn_XGmm_t.wav

Feature Extraction Implementation

The extract_features.py script:

1. Iterates through all WAV files

Output files:

4.3 Model Training Implementation

Data Preparation

`python

X = task_df[feature_cols].values # Shape: (n_samples, 88)

Cross-Validation Setup

`python

  - Stratified: maintains class balance in each fold

SVM Training

`python

Random Forest Training

`python

4.4 Feature Importance Implementation

Gini Importance

`python

rf_importance = pd.DataFrame({

Measures contribution to reducing impurity across tree splits.

Permutation Importance

`python

More robust: shuffles each feature, measures accuracy degradation.

4.5 Visualisation

`python

```
ax.barh(range(len(top20)), top20['importance'].values)
```

Horizontal bars for readability of long feature names.

4.6 Output Structure

`

figures/

Version Control

Git tracks code changes. Large files excluded:

`

4.7 Execution Workflow

`bash

2. Run analysis (~1 minute)

4.8 Summary

Implementation follows best practices:

 - **Modularity:** Separate scripts for extraction/analysis

All code available in project repository.

Estimated length: 4-5 pages