1 Appendix

A. Technical Details and Definitions

1. **Mel-Frequency Cepstral Coefficients (MFCCs):**

- **Definition:** MFCCs are coefficients that collectively represent the short-term power spectrum of a sound.
- **Application:** In this paper, MFCCs such as `mfcc_3`, `mfcc_11`, and `mfcc_5` were used to distinguish between HC and PD, with higher MFCC values often indicating PD.

2. **Jitter:**

- **Definition:** A measure of frequency variation from cycle to cycle in voice signals, indicating potential vocal fold instability.
- **Application:** Used to identify fine variations in vocal recordings that are symptomatic of PD.

3. **Shimmer:**

- **Definition:** A measure of amplitude variation from cycle to cycle used to detect issues in vocal fold function.
- **Application:** Used to identify fine variations in vocal recordings that are symptomatic of PD.

4. **Harmonic-to-Noise Ratio (HNR):**

- **Definition:** HNR is the ratio between harmonic components of the data and noise components. Lower HNR values indicate a breathier or noisier voice, which can be indicative of PD.
- **Application:** HNR contributed to the AI model's predictions by looking for conventional symptoms of PD.

5. **Fourier Transformation (FT):**

- **Definition:** A mathematical technique that transforms a time-domain signal into its constituent frequencies, providing a frequency-domain representation of the signal.
- **Application:** This study used FT to convert voice recordings into the frequency domain, aiding in accurately extracting key acoustic features.

6. **SHapley Additive exPlanations (SHAP):**

- **Definition:** Provides a way to explain the output of machine learning by showing the contribution of each acoustic feature to the predictions.
- **Application:** This study used SHAP to interpret the model's predictions, offering insight into the extent to which acoustic features contributed to the final diagnosis.

B. Model Architecture and Training Process

1. **MLP + CNN + RNN + MKL Learning Model Architecture:**

- This model combines MLP for non-linear data representation, CNN for local pattern recognition, RNN for temporal sequence analysis, and MKL for integrating multiple feature modalities.
- **Architecture Overview:**
  - **CNN Layers:** Extracted hierarchical feature representations from input spectrograms.
  - **RNN Layers:** Modeled the temporal dynamics of speech to identify sequential anomalies.
  - **MKL Layers:** Integrated diverse feature modalities, enhancing generalizability and robustness.
  - **MLP Layers:** Processes the combined representation to learn higher-level, non-linear relationships between acoustic features.
- **Training Process:** The model was trained using k-fold cross-validation to avoid overfitting and ensure quality output. The training involved multiple epochs, continuous monitoring, and tuning based on loss and accuracy metrics.

2. **Parameters:**

- **Learning Rate:** Adjusted based on training output to optimize model convergence.
- **Epochs:** An epoch is one complete pass through the entire training dataset. Training a model for multiple epochs improves performance by allowing it to learn patterns. This study trained AI models on a scale of 1-150 epochs with the flexibility to halt training at any point to prevent overfitting.
- **Batch Size:** All AI models were trained on the same 81 voice recordings to ensure consistency in input, thus minimizing confounding variables.

C. Data Preprocessing Steps

1. **Noise Reduction and Decibel Equalization:**

- Iyer, A. et al. (2023) removed all background noise from the 81 audio recordings. Furthermore, audio was equalized based on sex to maintain consistency across data [1].

2. **Handling Silent Intervals:**

- Iyer, A. et al. (2023) retained intervals of silence before and after the 81 audio recordings to preserve the natural speech patterns of the participants [1].

3. **Feature Extraction:**

- All audio files were processed using a Parselmouth library, a Python wrapper for Praat. Key acoustic features were extracted.

D. Supplementary Data and Visualizations

1. **Spectrograms:**

- HC and PD Spectrograms: Both spectrograms can be found as Supplementary Figures S2 online.
- Short-Time Fourier Transformation (STFT) Enhanced Spectrograms: Figures 6a and 6b are STFT-enhanced spectrograms, showing improvement in clarity and diagnostic utility.

2. **Evaluation:**

- Accuracy and Cross-Entropy Loss Metrics: This metric provides the accuracy and loss for each fold in the 5-fold CV, highlighting variability and areas for further model improvement.
  - An accuracy value above 80% indicates high-quality performance regarding the ratio of the number of correct predictions (both true positive and true negative) to the total number of predictions (true positive, true negative, false positive, and false negative).
  - A loss value below 20% indicates high-quality performance regarding predictions matching with true labels.
- The following is how the performance metrics were calculated:
  - **Accuracy:**
$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Instances}}$$
  - **Precision:**
$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$
  - **Recall:**
$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$
  - **F1 Score:**
$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$
  - **Confusion Matrix:**
$$\begin{bmatrix} \text{True Negatives} & \text{False Positives} \\ \text{False Negatives} & \text{True Positives} \end{bmatrix}$$

– The following table displays the average performance metrics across 5-fold CV of each model:

Model	Accuracy	Precision	Recall	F1 Score	Area under Curve (AUC)
MLP	48.22%	43.38%	28.00%	32.68%	0.4660
CNN	66.38%	63.46%	80.00%	70.05%	0.6625
CNN + MLP	58.15%	56.98%	73.00%	63.88%	0.5968
MKL + MLP	56.79%	56.76%	52.50%	54.55%	0.5674
RNN + MLP	61.46%	58.17%	80.00%	67.23%	0.6125
CNN + RNN + MLP	65.66%	63.83%	79.00%	68.95%	0.6599
MLP + CNN + RNN + MKL	91.11%	89.84%	92.50%	91.13%	0.9125

E. Code and Algorithm Explanations

1. **Pseudocode for Model Training:**

- Model Workflow:

```
Input data is fed into CNN layers

CNN Layer:
for each convolutional layer:
    apply convolution and pooling
Output: Feature maps

Pass feature maps to RNN layers

RNN Layer:
for each recurrent layer:
    process temporal sequences from feature maps
Output: Temporal feature representation

Pass outputs from CNN and RNN layers to MKL

MKL Layer:
    learn a kernel-based representation from both CNN and RNN outputs
Output: Combined representation

Pass combined representation to MLP layers

MLP Layer:
for each fully connected layer:
    process combined data to learn non-linear relationships
Output: Refined representation

Fully Connected Layers:
    pass the refined representation to dense layers for further processing
Output: Prediction distribution

Output Layer:
    provide the classification output
```

This pseudocode outlines the primary steps in training the hybrid MLP + CNN + RNN + MKL model.

F. Dataset Information

1. **Source:**

- The dataset used in this study was obtained from Figshare, titled *"Voice Samples for Patients with Parkinson's Disease and Healthy Controls"* (`https://figshare.com/articles/dataset/Voice_Samples_for_Patients_with_Parkinson_s_Disease_and_Healthy_Controls/23849127`).

2. **Participant Details:**

   - 81 voice recordings (41 from HCs and 40 from PD patients).
   - Demographic information, including sex ratio, age at collection, Hoehn & Yahr stage of PD, and length of disease, can be found in *Table 2a*.

3. **Ethical Considerations:**

   - All data was from a public dataset and were anonymized, thus adhering to the ethical standards of data usage and participant privacy.