**Report on Speech Command Recognition Project Using ResNet Model**

**MID SEMESTER LAB EVALUATION**

**CONVERSATIONAL AI: SPEECH PROCESSING AND SYNTHESIS(UCS749)**

Submitted by:

Bhavika Jindal

102103628

4CO23

BE Fourth Year, COE

Submitted to:

Dr. Raghav B.Venkataramaiyer



**Computer Science and Engineering Department**

**Thapar Institute of Engineering and Technology, Patiala**

**September 2024**

**Summary of Research Paper Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition:**

The paper **"Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition"** by Pete Warden introduces a dataset designed to train and evaluate keyword spotting systems. It focuses on recognizing small sets of words, useful for voice-activated interfaces. The dataset contains over 105,000 utterances of 35 words spoken by 2,618 speakers. The paper also provides baseline performance results and discusses the importance of efficient on-device speech recognition, emphasizing the need for smaller, energy-efficient models due to the constraints of mobile and embedded devices. The dataset aims to standardize and promote reproducibility in the field.

**1. Clarity of Thought Process and Presentation**

The code demonstrates a logical and well-organized approach in designing and implementing the speech command recognition project. It clearly outlines the steps, from data acquisition, feature extraction, and model building to training and saving the final model. Inline comments throughout the code aid in understanding each section, contributing to the overall clarity and flow of the project.

**2. Data Processing Skills**

The data processing skills are evident, particularly in how the speech dataset is handled. Key steps include:

* **Data Acquisition**: The dataset is downloaded and extracted efficiently, with checks in place to prevent redundant downloads.
* **Feature Extraction**: The librosa library is used to extract Mel-Frequency Cepstral Coefficients (MFCCs) from the audio files, with padding or truncation applied to ensure uniform length across all samples.
* **Label Handling**: Labels are correctly extracted from file paths, and only valid labels are retained. The LabelEncoder is used to convert labels into a format suitable for model training.

**3. Model Fine-Tuning/Training Skills**

The use of a ResNet50 model, although unconventional for speech recognition, reflects the ability to adapt pre-existing models to new tasks. Notable aspects include:

* **Model Construction**: A ResNet50 model is constructed, and input data is resized to match the expected dimensions of the network.
* **Layer Customization**: Additional dense layers and dropout are incorporated to prevent overfitting and improve generalization.
* **Model Training**: The model is trained using sparse\_categorical\_crossentropy as the loss function and the adam optimizer, showing a solid understanding of model training requirements.

**4. Details of Progress: Problems Encountered and Solutions**

Several challenges were likely encountered during project development, with clear solutions provided:

* **Challenge**: Handling varying lengths of MFCC features.
  + **Solution**: The features were padded or truncated to ensure uniformity across all samples.
* **Challenge**: ResNet50 requires 32x32 input images, which differs from the MFCC feature shapes.
  + **Solution**: A resizing layer was added to adjust the MFCC features to the correct dimensions.

**5. Adaptability of the Pipeline**

The pipeline shows a high degree of adaptability. The key points include:

**Feature Extraction:** The `load\_audio\_file` and `pad\_features` functions are designed to be flexible, allowing easy modifications for different types of audio features or different datasets.

**Model Architecture:** The use of a pre-trained model like ResNet50, with layers added or modified, means that this approach can be easily adapted to other types of speech commands or even different domains such as image or video processing.

**6. Scalability of the Approach**

The pipeline demonstrates a high degree of adaptability, with key points including:

Feature Extraction: Functions like load\_audio\_file and pad\_features are flexible, allowing for easy modifications if needed for different types of audio features or datasets.

Model Architecture: By utilizing a pre-trained model like ResNet50 with additional layers, the approach can be easily adapted to other speech command recognition tasks or even different domains such as image or video processing.

**Strengths and Shortcomings**

* **Strengths**:
  + Employing a robust, pre-trained model (ResNet50) known for handling complex tasks.
  + Flexible, modular code that allows easy adaptation and scalability.
  + A clear, structured approach to data handling and model training.
* **Shortcomings**:
  + ResNet50, while powerful, may not be the optimal choice for speech recognition tasks. CNN architectures designed specifically for audio processing could yield better results.
  + Resizing the MFCC features to fit ResNet50 may result in a loss of important spectral information.
  + Model training could be further enhanced with learning rate scheduling, data augmentation, or transfer learning from a pre-trained audio-specific model.

**Summary:**

The project successfully employs a ResNet50 model to achieve over 87% accuracy in recognizing speech commands. The pipeline is well-structured, adaptable, and scalable, though it could benefit from leveraging audio-specific models to further improve performance.