1. **Dataset and Preprocessing**

* **Dataset Information**:
  + Total dataset size: **575,912 samples (approx. 10110 samples per class)**
  + Number of classes: **57 (Vowels = 12, Consonants = 35, Digits = 10)**
  + Data format: **Images with corresponding labels**
* **Preprocessing Techniques**:
  + **Converted the images to greyscale**
  + Conversion to tensors
  + Normalization to a range of **[-1, 1]**
  + Label encoding using argmax for categorical labels
* **Data Splitting**:
  + Train-validation split: **80-20** (using train\_test\_split)
  + Batch size: **32**
  + Use of DataLoader for efficient data handling with **8 worker threads**

1. **Model Architecture**

* **Convolutional Neural Network (CNN) Design**:
  + 3 Convolutional layers with batch normalization
    - 1st Conv layer : 1 -> 32
    - 2nd Conv layer : 32 -> 64
    - 3rd Conv layer : 64 -> 128
  + Kernel size: **5×5** in each conv layer.
  + Batch Normalization in the conv layers.
  + Activation function: **ReLU**
  + **Max pooling (2×2)** after each convolution with stride = 2.
  + **Dropout (0.5)** for regularization in MLP.
  + Fully connected layers:
    - **2048 neurons → num\_classes (57 classes)**
* **Parameter Initialization**:
  + Dynamic computation of self.\_to\_linear for correct input size to the fully connected layers

1. **Training Procedure**

* **Hardware Used**: Mention whether training was done on **GPU or CPU**
* **Loss Function**: **CrossEntropyLoss**
* **Optimizer**: **Adam** (learning rate = **0.0005**, weight decay = **0.0001**)
* **Number of Epochs**: **100**
* **Performance Metrics Tracked**:
  + **Training Loss & Accuracy**
  + **Validation Loss & Accuracy**

1. **Model Evaluation and Results**

**Evaluation Metrics:**

* **Accuracy:** 92.38% for training and 91.39% for testing.
* **Loss:** 0.2183 for training and 0.2437 for testing
* **Confusion Matrix:** A confusion matrix is used to visualize the classification performance across different classes. This helps in identifying misclassifications.
* **ROC Curve:** The ROC curve provides insights into the model’s performance at different classification thresholds. AUC scores for each class can be computed to compare effectiveness.
* **Precision-Recall Curve:** Given potential class imbalances, the Precision-Recall curve offers additional insights, especially for underrepresented classes.