

Speech Understanding Programming Assignment - 1

Instructions

1. All code needs to be written with Python + PyTorch only.
 2. Create a repository on your Github. Push this Assignment with all of your codes, report and readme file on that.
 3. You need to submit a single zip file containing the report, codes and readme file. The zip file should be named as Rollno PA1.zip.
 4. In the report and in the private comment of your submission on the Google Classroom, you have to mention the link to your Github repository.
 5. Do cite the libraries/codes/references in the report.
-

Question 1: In a group of two students, you have to perform a detailed analysis of one task related to speech processing (examples include speech recognition, speaker identification, emotion recognition, speech synthesis, accent detection, or any other relevant task). Perform a **detailed analysis** covering the following aspects: Prepare a detailed report and a short presentation by comparing and briefly describing the SOTA models available for each task. These should include the following:

- Explain the task and its importance in the real world.
- Analyze the strengths and limitations of state of the art models or tools in terms of the methods or models available.
- Discuss the results in terms of the metrics used to evaluate the task, including their strengths and limitations.
- Suggest what are the open problems and opportunities corresponding to that problem statement.
- Submission should include the presentation, report and the codes/datasets you have worked with for this question.

Note: Not more than 3 groups can select the common tasks. You have to fill in the group details and tasks your group will choose by tomorrow, in the sheet link given below. The sheet will be filled on a First Come First (FCF) basis.

Sheet link: [📄 Speech Understanding2025 - PA1](#)

Question 2: This question is to be submitted individually. Experimenting with Spectrograms and Windowing Techniques.

Task A.

1. Use the UrbanSound8k dataset for this assignment. Download the dataset from - <https://goo.gl/8hY5ER>

2. Understand and implement the following windowing techniques:
 - a. Hann Window
 - b. Hamming Window
 - c. Rectangular Window
3. Write a Python program to apply the above windowing techniques → Generate spectrograms using the Short-Time Fourier Transform (STFT)
(Compare the spectrograms visually and analyze their differences. Discuss the correctness of windowing performed.)
4. Train a simple classifier (e.g., SVM or neural network) using features extracted from the spectrograms and evaluate the performance results comparatively in different techniques.

In Report write the comparison and analysis of the results computed.

Task B.

Select 4 songs from 4 different genres and compare their spectrograms. Analyze the spectrograms and provide a detailed comparative analysis based on your observations and speech understanding.

Submission for both the tasks includes codes, songs you have used for analysis, and the report.