Course Project

Code of Honor. All external resources used in the project, including research papers, open-source repositories, datasets, and any content or code generated using AI tools, e.g., ChatGPT, GitHub Copilot, Claude, Gemini, must be *clearly cited* in the final submission. The final report must also include *a clear breakdown of individual group member contributions.* Any lack of transparency in the use of external resources or in reporting group contributions will be considered academic dishonesty and will significantly impact the final evaluation.

Topic Automatic Speech Recognition (ASR)

Objective Design and implement an Automatic Speech Recognition (ASR) system that converts spoken audio into text, with a primary focus on batch transcription of pre-recorded audio files. We then aim to evaluate the impact of various components of ASR systems on transcription accuracy and generalisability.

Motivation Automatic Speech Recognition is a foundational problem in deep learning with wide-ranging real-world impact. In education, for example, accurate lecture transcription can reduce cognitive load, improve review efficiency, and enable new modes of interaction with learning material [1].

Beyond its practical value, ASR offers a rich learning opportunity due to the many complex components involved in its implementation, ranging from audio signal processing and feature engineering to layered modeling approaches that combine multiple interdependent models to effectively learn the diversity and temporal structure of speech [2].

Requirements

1. Implementation
   - Implement pipeline to compress raw audio data to spectrograms.
   - Implement a baseline deep neural network model from scratch leveraging the DeepSpeech ASR architecture [3].
2. Evaluation
   - Analyse the impact of hyperparameters and modifications/extensions on model performance, and justify the final choice of parameters and model.
   - Evaluate the performance of the model on different datasets, and against publicly available ASR models.
3. Potential improvements
   - Identify the limitation(s) of the chosen architecture.
   - Investigate baseline enhancement using a pre-trained language model.

Milestones

1. Literature review
   - ASR architectures used in industry.
   - Audio datasets suitable for training on limited compute.
   - Metrics for evaluating ASR model performance.
2. Implementation & Testing
   - Data preprocessing pipeline.
   - Training and testing pipeline.
   - Baseline ASR model.
   - Iterative improvement to baseline model.
   - Generalisation to other datasets.
3. Evaluation & Analysis
   - Effects of modifications/extensions to baseline model.
   - Compare with publicly available models.
   - Perform ablation experiments (stretch goal).
4. Final Presentation and Report

Submission Guidelines The main body of work is submitted through Git. In addition, each group submits a final paper and gives a presentation. In this respect, please follow these steps.

- Each group must maintain a Git repository, e.g., GitHub or GitLab, for the project. By the time of final submission, the repository should have
  - Well-documented codebase
  - Clear `README.md` with setup and usage instructions
  - A `requirements.txt` file listing all required packages or an `environment.yaml` file with a reproducible environment setup
  - Demo script or notebook showing sample input-output
  - *If applicable,* a `/doc` folder with extended documentation
- A final report (maximum *5 pages*) must be submitted in a PDF format. The report should be written in the provided formal style, including an abstract, introduction, method, experiments, results, and conclusion.
  Important: Submissions that do not use template are considered *incomplete.*
- A 5-minute presentation (maximum *5 slides including the title slide*) is given on the internal seminar on Week 15, i.e., *Dec 8 to Dec 12,* by the group. For presentation, any template can be used.

Final Notes While planning for the milestones please consider the following points.

1. You are encouraged to explore innovative approaches to conditioning or generation as long as the core objectives are met.
2. While computational resources are limited, carefully chosen datasets and training setups can make even diffusion models feasible. Trade-offs, e.g., resolution, training steps, are expected and should be justified.

3. Teams are expected to manage their computing needs and are advised to perform early tests to estimate runtime and training feasibility. As graduate students, team members can use facilities provided by the university, e.g., ECE Facility. Teams are expected to inform themselves about the limitations of the available computing resources and design the model accordingly.

References

[1] Grand View Research. Speech-to-text api market size, share & trends analysis report by component (software, services), by development, by organization size, by application, by verticals, by region, and segment forecasts, 2025–2030. Technical report, Grand View Research, 2025. Report ID: GVR-4-68039-963-7. Accessed: 2026-02-05.

[2] Md. Nayeem, Md Shamse Tabrej, Kabbojit Jit Deb, Shaonti Goswami, and Md. Azizul Hakim. Automatic speech recognition in the modern era: Architectures, training, and evaluation, 2025.

[3] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, and Andrew Y. Ng. Deep speech: Scaling up end-to-end speech recognition, 2014.