Alexandria University
Faculty of Engineering
Computer and Systems Engineering
Department

CSE: Pattern Recognition
Assigned: Mon, May. 19, 2025
Due: Tue, Jun. 26, 2025

Assignment #4: Environmental Sound Classification

1 Problem Statement

Environmental sounds are critical cues for scene understanding in real-world applications like surveillance, autonomous navigation, and assistive technology. In this assignment, you will implement two deep learning approaches for classifying environmental audio recordings using the UrbanSound8K dataset. The first model will be based on Recurrent Neural Networks (RNNs), and the second will be a Transformer-based model, which you will implement from scratch (i.e., no use of nn.Transformer or Hugging Face models).

2 Download the Dataset and Understand the Format

1. We will use UrbanSound8K dataset that is available at the following link:

   https://urbansounddataset.weebly.com/urbansound8k.html

2. Explore the dataset and understand its structure. It consists of:

   (a) 8732 labeled sound clips ($\leq$ 4s long)

   (b) 10 class labels: air_conditioner, car_horn, children_playing, dog_bark, drilling, engine_idling, gun_shot, jackhammer, siren, street_music.

3. Write your own function to:

   (a) Load audio samples.

   (b) Plot the waveform and/or spectrogram.

   (c) Listen to examples from different classes to build intuition.

4. The dataset consists of 10 folds; we will use

   (a) first 6 folds for training.

   (b) 7th and 8th folds for validation.

   (c) 9th and 10th folds for testing.

3 Create the Feature Space

Analyze different features you can extract and create your own feature space. You can use these, but we encourage you to search for others.

1. Melspectograms

2. Energy
3. MFCCs

4 Building the Model

4.1 RNN Model

Implement an RNN-based classifier using one of the following

1. RNN
2. LSTM

4.2 Transformer Model

Build a Transformer-based classifier from scratch. You must implement the following blocks manually:

1. Multi-Head Self Attention
2. Position-wise Feedforward Network
3. Positional Encoding
4. Layer Normalization and Residual Connections
5. Use your implemented encoder block to build the classifier.

Do not use nn.Transformer, transformers, or similar libraries.

5 Big Picture

Compare the performance of the learned models (Different features, different learning models, hyperparamater) by realizing the following.

1. Compute the accuracy and F-Score for each model.
2. Plot the confusion matrices and find the most confusing classes.

6 Bonus

- Design and implement a novel model architecture (e.g., CNN-LSTM) and evaluate its performance in comparison with existing models.
- Top 3 students on the leaderboard.

7 Important Notes

1. Start working on the assignment early, as training takes time, and there will be no postponements.
2. Be innovative, do your research, and do your best.

8 Submission Notes

- Work in groups of 3 students.
- Submit a **Jupyter Notebook** or a well-commented Python script containing your code.
- Submit a **PDF report** detailing your approach, results, and analysis.
- Ensure reproducibility by setting random seeds and structuring code properly.

Note

- While discussions are allowed, all code and reports must be your own work. Plagiarism will result in a penalty.