CS F425 Deep Learning
2nd Semester 2023-24
Term Project – Task 1
Sound classification

The term project (task 1) for the Deep Learning course will be a team activity. The recommended size is THREE. Larger teams will not be permitted.

Description

As a part of the task 1 for your term project, you will build CNN based classifiers for classifying audio samples. Most of the concepts required for this project have been covered in the class. However, you are recommended to go beyond the classroom knowledge and try to build an interesting and innovative model by exploring more concepts yourself.

Task

You are being provided with training and validation datasets containing 13 different audio sounds such as laughter, car horn, dog bark, etc. Each sound corresponds to one class. There are 4861 samples in the training set and 1209 samples in the validation set.

Although audio data, in general, has a temporal factor associated with it, your task does NOT require you to treat this as a time-series/sequence modelling problem.

Your task is to utilize this sound data to create CNN-based models and perform multi-class classification for these 13 sounds. *It is left up to your creativity that how you utilize audio signals in a CNN-based architecture which is typically meant for images.* Try to be innovative and novel here because we will give weightage to that in the evaluation!

You need to train **TWO** different **CNN based architecture** (of your choice) for this multi-class classification task. It is strongly recommended that the two architectures that you choose are quite different from each other and are NOT minor variations of each other.

Performance metrics

The performance metrics for task 1 are accuracy, precision, and recall (class-wise and overall).

Evaluation

Although the task 1 is being conducted similar to a challenge or a hackathon, the evaluation will not solely be on the performance of your model. The performance metrics mentioned above will carry some weight in the evaluation. But more importantly, you will be evaluated on the *choice* of your network design (how interesting, novel or innovative it is), thorough experimentation and comparison of both architectures, and your understanding of what-you-did and why-you-did.

Important notes:

1. Your architecture MUST utilize CNNs. However, use of other concepts from deep learning, or even from areas of machine learning or signal processing (in addition to CNNs) which add novelty to your work is very much welcome.
2. You should ONLY use the training and validation datasets provided, and not use any other datasets for training. This is important because the dataset has been curated from multiple online sources. If you add more training data, you run the risk of over/under-fitting.
3. Your models will be tested on a separate test dataset which will NOT be shared with you. You will submit the trained model to us and we will evaluate your model's performance on the test set.
4. Use of pre-trained models (transfer learning) is strictly forbidden. All models must be trained from scratch.
5. Use of training hacks, data augmentation approaches, etc. is encouraged. However, all such techniques must be properly documented and you should explain why they were used and what benefit you got by using them.
6. Choose your CNN architectures wisely. You should be able to explain and justify your architecture and all its design choices. You are allowed to use existing code (from Github or other places) but with proper attribution (and proper motivation – why you used that particular architecture for this task). You should also remember that you need to train/test/demo your code on the free version of Colab, and a heavy architecture might not train easily on the free version of Colab.

Submission

- You will need to submit your code along with the final weights (i.e., the trained model).
- You also need to submit a brief/concise report which provides all details of the two architectures chosen by you, the training process, and any other details related to your implementation.
- You must explain all the training or design choices adopted by you.
- Report the performance (as per performance metrics) you obtained for both the tasks.
- You should also report the time for training and testing, the number of parameters in your model, the choice of hyper-parameters, and loss-accuracy curves.

Final evaluation may also involve a viva or something similar. More details will be provided in due course of time.

Deadline

23rd March 2024 midnight.