PAT 498/598: Special Topics: Generative AI for Music and Audio Creation (Fall 2024)

Assignment 2: Musical Note Classification using CNNs

Due at 11:59pm ET on October 7

1 Musical Note Classification

1.1 Dataset

We will be using the NSynth dataset (Engel et al., 2017). NSynth is a collection of 305,979 audio samples of single-shot musical notes produced from 1,006 commercial sample libraries. While NSynth provides some other metadata, in this assignment, we will only be using the instrument family annotation as our classification target. Here is the list of instrument families in NSynth.

0	bass	4	keyboard	8	string
1	brass	5	mallet	9	synth_lead
2	flute	6	organ	10	vocal
3	guitar	7	reed		

1.2 Task

Your task is to train a convolutional neural network (CNN) that can classify the mel spectrogram of an audio file into its instrument family. To save the computation time, we will use a 64x64 mel spectrogram by downsampling the audio to 8kHz and computing the mel spectrograms using an FFT size of 2,048, a hop size of 512 and 64 mel bins. The original and processed NSynth can be found at the following directory on Great Lakes.

- NSynth dataset: `/nfs/turbo/smtd-hwdong/data/nsynth`
- Preprocessed dataset: `/nfs/turbo/smtd-hwdong/data/nsynth/processed` (where you can find the precomputed 64x64 mel spectrograms)

(Another version of the preprocessed dataset with 128x128 mel spectrograms can be found at `/nfs/turbo/smtd-hwdong/data/nsynth/processed_128x128`, where the mel spectrograms are computed on 16kHz audio with 128 mel bins, FFT and hop sizes unchanged.)

You will want to copy the dataset to your home directory to speed up the data loader.

```
# Create a data directory in your working directory (i.e., where the
# Jupter notebook is)
mkdir data
mkdir data/nsynth
cd data/synth

# Copy the zipped datasets (this will take a while)
cp /nfs/turbo/smtd-hwdong/data/nsynth/processed/train.zip .
cp /nfs/turbo/smtd-hwdong/data/nsynth/processed/valid.zip .
cp /nfs/turbo/smtd-hwdong/data/nsynth/processed/test.zip .

# Unzip the data files (this also takes a while)
unzip train.zip
unzip valid.zip
unzip test.zip
```

You can find a sketch Jupyter notebook here. With the provided sketch code, please complete the followings:

- Train the baseline model
  - Plot the training and validation loss curves
  - Report the test loss and accuracy
- Experiment two (or more) of the followings to see if you can improve the performance
  - Use other CNN architectures (e.g., number of layers, kernel sizes, strides, pooling layers)
  - Use other activation functions such as sigmoid and tanh functions
  - Use other optimizers such as SGD with momentum, Adagrad, Adadelta, RMSProp and Adam
  - Use different batch sizes
  - Apply regularization methods such as dropout and L1/L2 regularization
  - Apply normalization methods such as batch and layer normalization
  - Use 128x128 mel spectrograms as inputs, where the preprocessed dataset can be found at `nfs/turbo/smtd-hwdong/data/nsynth/processed_128x128`
  - Any other extensions!
- Analyze and discuss the experimental results

You work will ***not* be graded by the performance of your final model,** but rather **the amount of work you put in exploring different techniques and analyzing the experimental results**. Thus, please also report any negative results that you find not working and discuss why it is not working.

Please submit your code and a short report that summarizes the experimental design and your findings. The report should be no more than 2 pages, excluding references, and you may use any template you like. You will receive zero credit if the code is missing.

1.3 Rubrics

- Training of the baseline model (4pt)
- Experimental results (3pt)
- Analysis and discussions (3pt)

1.4 Computing Resource

You will want a GPU for efficient training (easily 10x or more speedup for deep learning models). You will be provided 3000 CPU hours in total on Great Lakes for the whole semester, where the quote is for all the assignments and final project. Please refer to this page for more information for working with Great Lakes. You may also find Google Colab useful.

2 Submission

- All assignments must be completed on your own. You are welcome to exchange ideas with your peers, but this should be in the form of concepts and discussion, not in the form of writing and code.
- Please provide proper citations/references for any external resources you use in your writing and code.
- Please submit your work to Gradescope.
- Late submissions will be deducted by **3 points per day**.

References

[1] Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Douglas Eck, Karen Simonyan, and Mohammad Norouzi, "Neural Audio Synthesis of Musical Notes with WaveNet Autoencoders," *ICML*, 2017.