FACULTY OF COMPUTER SCIENCE

General Purpose Audio Tagging

Submitted by	Balaji Dhakshinamoorthy. B00777437.

Table of Contents

Abstract:

Kaggle General Purpose Audio Tagging is a Sound event classification project with the objective of recognising sound labels for the diverse and broad categories of environmental sounds.It is the second challenge of DCASE Community with the challenge of tagging sound events.The audio files are classified through the combination of audio preprocessing technique called MFCC and Convolution 2D layers. Thus the designed model produce considerably good results even in the smaller subsets of original audio duration with the public usage label accuracy of around 64 percentage for top label prediction.

Introduction:

This Project is about building a General Purpose Automatic audio tagging classifier which contains audio of broad and diverse categories like musical instrument,animal sounds and human sounds etc. The sound labels have some varying reliability with some of them are manually annotated and rest of them are freesound[2] generated automatic labels.The project main task is to classify the audio into one of the 41 different classes comprising of diverse categories. This Project is hosted in the kaggle a platform for hosting machine learning competitions with various community of people. This project has a scope for the application in automatic video description and also for detecting acoustic events.

The model designed for this model produces pretty much decent results even with constrained input audio duration of around 2 seconds and shallow convolution 2D layers.The model considers the preprocessing technique of audio called

MFCC[7].The preprocessing stage is done with the help of library called librosa [6]. If modified with more deep neural networks and with the combination of audio preprocessing like Mel spectrogram with 128 Mel Scale with full audio input resolution it has the capability of producing much better annotation of test labels.In terms of accuracy it matches closely with the accuracy produced by the baseline model [3] even with shallow neural networks. So it has much better scope to produce good results if hypertuned with right mix of filter channels and optimal audio input resolution and deep neural layers.

Related Works:

There are good number of previous works constructed on the basis of MFCC and Convolutional neural Networks. The baseline model[3] uses the shallow Convolutional neural networks which is a scaled down version of general Deep network used in computer vision domain.They used the log mel spectrogram and three convolutional neural networks with 100,150 and 300 channels to classify audio labels. This model produced MAP score of around 70 %. Keunwoo Choi et all [4] proposed a convolutional recurrent model for music classification.They used convolution layers for music feature extraction and applied gated recurrent network on top of them to produce the summarisation of those features.They also used spectrogram as audio feature input and showed better results in terms of music classification than the models which used only convolutional neural network.

Mingi yeom at all[5] proposed a model for the general purpose audio tagging project using 5 convolutional 1d networks and mel spectrogram using 128 mel scales and produced mean average precision(map) of around 71 percentage

which is quite similar to the accuracy produced by the baseline system[3]. My proposed model is similar to the above mentioned models in the context of preprocessing audio signals and convolutional layer approaches but with the variation of MFCC and Convolution 2D kernels respectively. The model produces similar results to the aforementioned models even with constrained audio input resolutions.

Dataset:

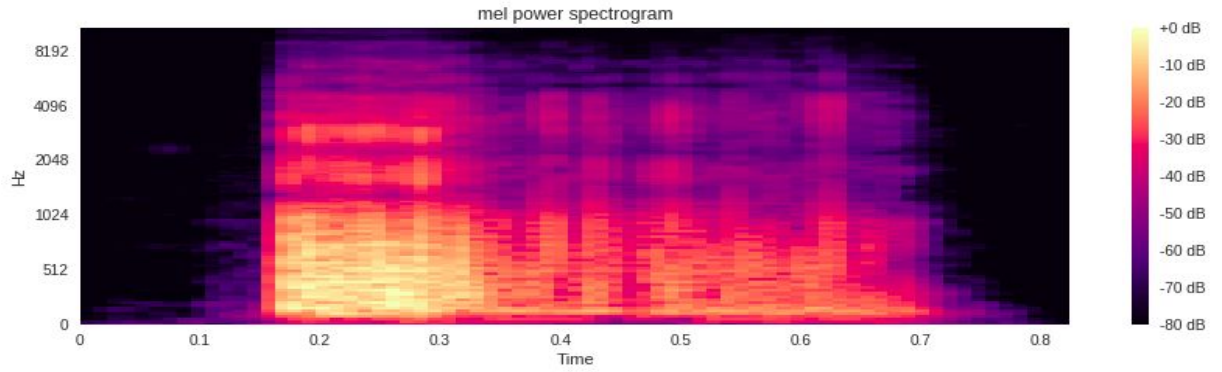
The dataset is from Kaggle website which was hosted as General Purpose audio tagging competition[3] from March 30th 2018 to July 1 2018. The community collected all of the audio samples from the Free sound [2] which is a sound sharing website.The website hosts 38000 sounds uploaded by its users.It has heterogeneous sounds which has diverse categories of sound events.They are released by the creative commons license which facilitates sharing of those resources for the research purposes.Those sound files are annotated by the users who upload them by tagging their names and description etc.

FSD kaggle generated automatic labels for some of those sound files by matching a set of declared audio labels with each of the tagged audio files submitted by the users.Since they pick the labels just from the candidate tagging they are called weak labels.There is a another process of validation task where they manually verify those annotated tags and declare the right label for those sound events. So this dataset has some of the samples which are manually annotated and

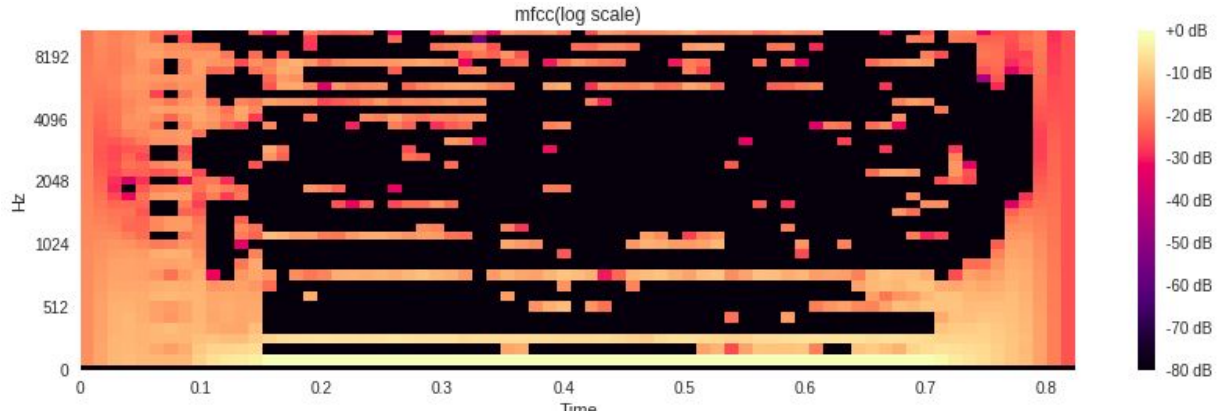
the rest are Free sound generated labels which are also called weak labels. This brings in varying reliability factor for each labels of the training samples.

The dataset has 9473 training samples with a mix of manually annotated and freesound generated label samples.The test dataset has 9400 samples.1600 test samples are manually verified samples with the similar distribution as in training samples.The test set is complemented with 7800 padding sounds which will not be used for scoring systems. Approximately 19% of the 1600 test samples are treated as Public usage which is used for verifying live competition results and the remaining 81% of the files are considered as private usage.The prediction is evaluated by using Mean average precision(Map) technique by matching true label with any of the top three softmax predicted label for each test sample.In terms of credit, match of true label with either second or third prediction will be given slightly less weightage than the top softmax prediction.

The dataset has audio files, each of varying duration.But for implementing CNN on top of the model there is a need for the uniform input dimensions.The sampling rate of each audio file is 44100 Hz.The model considers audio duration of just 2 seconds similar to the approach mentioned in the kernels of kaggle [1]. This way it either zero pads the small duration audio files or it extracts the audio files of specific duration from the large duration audio files.Then the audio Preprocessing techniques need to be applied on top of each audio file.The preprocessing Makes sure that height of all images remains same and the fixed audio duration retains the same width dimension for all audio samples.

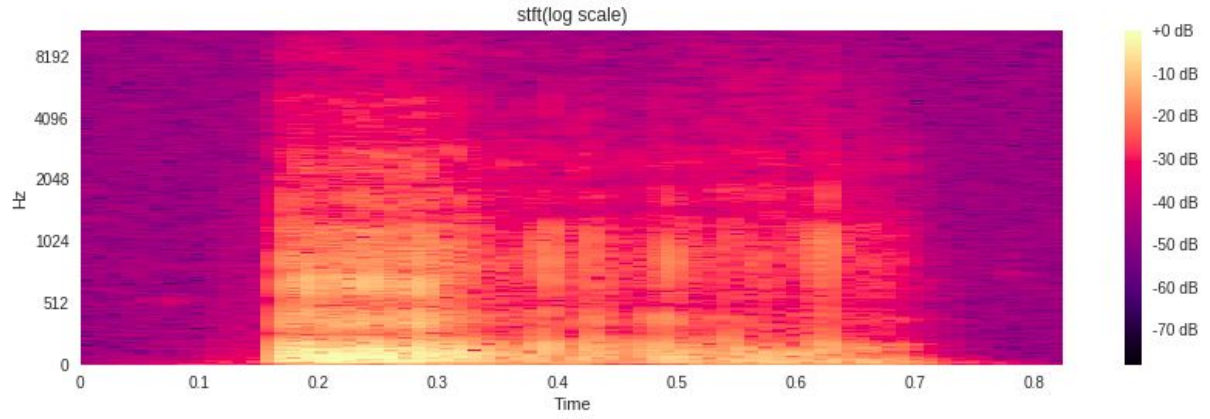


It is a Mel Spectrogram image of audio file with 128 mel scale.It is the data generated by the Mel Spectrogram of the librosa which is generated by taking a fourier transform of the signal.The second variant is MFCC also called Mel Frequency Cepstral Coefficient with N_mfcc=40 bins.It is basically obtained by taking the discrete cosine transform of the log power spectrum of the Spectrogram.



As seen from the above diagram,MFCC extracts information about the feature of the audio with low dimension (40X173) and their correlation can be learned by passing them on to the deep neural networks.There is a another variant called Short

time fourier transform as shown below.



It has much better capability of feature extraction by dividing long time frame of audio into several small frames and applying fourier transform on top of them to produce feature images.They produce higher dimension images than previous models so it requires more deeper neural network models to learn the pattern of the audio signals.Since my model has restriction in terms of resources I decided to go for MFCC for audio preprocessing.

Methods:

The Approach to classify the model is based on constructing Convolution 2D on top of the MFCC audio feature images.The model has around 4 convolution 2D layers each with same channels and kernel size and one feed forward layer with RELU activation function before passing them on to the softmax layer.Each Convolution layer has Batch normalization and dropout attached with it to avoid overfitting.This model was decided because of the Resource constraints. Originally, the plan was to use the raw audio waveform using wavenet [9] or convolution 1d with feed forward layer to predict the labels. But, the Wavenet Plan

was dropped because of the difficulty faced in implementing the alternative discriminative approach of the work. And also since the audio dimension varies a lot it required a much deeper system like Multi Layered RNN to predict the labels. Because of the resource constraints, there was also a problem in loading the entire training audio files, particularly if the audio preprocessing produced much higher dimension.So MFCC was approached with 40 bins which produced dimensions of around 40X173 for all audio files by considering just two seconds of Audio duration.The Advantage of the model is it takes less time to preprocess the data and produces good training results but the disadvantage is it fails to generalize much better because of the low audio duration and lack of deep layered networks.The model is listed below

Model Layers and Functions	Number of Channels x Kernel size or Dense Layers
Conv 2D ,Batch Normalization,Dropout(0.1)	32 x (4,10)
MaxPooling2D	2x2
Conv 2D ,Batch Normalization,Dropout(0.1)	32 x (4,10)
Max Pooling 2D	2x2
Conv 2D ,Batch Normalization,Dropout(0.1)	32 x (4,10)
Max Pooling 2D	2x2
Conv 2D ,Batch Normalization,Dropout(0.1)	32 x (4,10)
Max Pooling 2D	2x2

Feed Forward layer Dense,RELU	64
Softmax	41

This model configuration is similar to the article published in the kernel [1].The model produced some decent results when compared with baseline model and other similar related works.The baseline model [3] configuration is listed below.

Model,Layers and Functions	Channels x Kernel size x Dense Layers
Input	25 x 64 x 1
Conv 2D	100,(7x7),1
Max Pool 2D	(3x3),(2x2)
Conv2D	150,5x5,1
MaxPool 2D	(3x3),(2x2)
Conv 2D	200,(3x3),1
Global Max Pooling	
Softmax	41

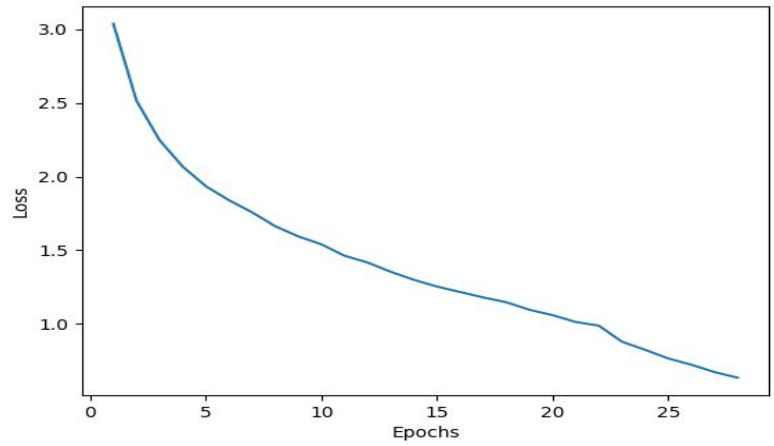
As seen from the above configuration the baseline model[3] is similar to our model in the approach of restricting the input size and the number of convolutional networks and also in terms of similar levels of test accuracy. So it is pretty evident that there is a need to process more frames of the input audio duration and more deep neural layers and ensembled models to produce better results.

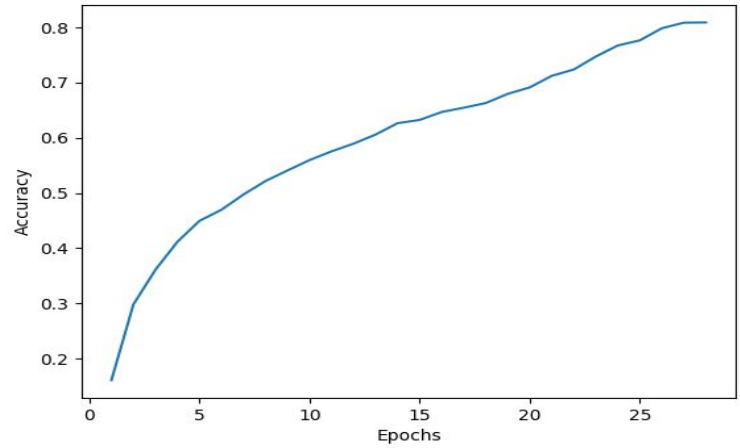
The easiest part of implementing the work is model implementation using keras [10].The hardest part with the model is preprocessing entire audio input and larger training time requirement to test the model.So even if there is a small changes in the model it takes more time to predict the efficiency of the model with limited resources. It pretty much had a big impact in trying out some deep convolutional neural networks and bigger channels which would have produced better results.

Experiments:

I started the experiment with the spectrogram of 128 mel scale and the input audio dimension of 128x1400 after running some experiment to find the average of top 20 mode of the training audio samples duration.But it requires a lot of computation power to process the audio.So I decided to switch to MFCC with 40 bins and carried out the experiment with some 2 layer cnn and 40 fixed channels.The computation took around 1 hour of training on Google Collab and they produced results of around 50% accuracy on private and public sets respectively.Then I reconstructed my model based on some help from kernel[1].

So after some more trials, I fixed my model parameters close to the model suggested by the kernel[1].It produced better results this time with considerably low loss and more training accuracy.Training accuracy shoots to 90 percentage in 50 epochs.But when tested for the validation accuracy by considering the last 1473 training sample as validation set it produced accuracy of around 63 percentage.The results are displayed below.





I performed further experiments with some LSTM layers on top of the model.But after adding one LSTM layer with 128 nodes and 10 input time steps it adds considerable amount of load time on computation and took more than 2 hours to run the code and after that they considerably produced better training results but not big improvement in validation accuracy. If there is no Computational Complexity, I might have tried with more channels for each convolution layers and with at least two bidirectional LSTM or GRU for further evaluation along with the increased duration of the input audio files to further validate the results. I infer that it might have played a big part in analysing the pattern of the audio. And also there

should be some pruning in input training data which has some non verified labels so there might be a big chance for that added noise to play a key role in not making the model to generalize better with shallow neural networks.This presumption will be clear if further experiment with same configuration but with more added layers starts producing better results.

In order to compute the accuracy of the model, private and public usage labels from test_post_competition.csv is extracted for model evaluation.The Map score is not tested for the model.Instead the model accuracy is based on matching top predicted label to true test label.The model produced around 64 percent accuracy.since Map considers two more top softmax outputs for each prediction the model accuracy is assumed to be quite similar in terms of map accuracy produced by the baseline model[3]**.**The model also has some overfitting problem since it produces better training results like close to 95% and not so great test results.Despite applying batch normalization and regularization technique like dropout there is not much increase in test results.It is also because of my lack of experiments with more deep convolution networks or channels.Increase in Convolution channels along with the Increased input Audio duration will help model to generalise better.In terms of optimization Adam with default parameters produced better results than SGD and Adadelta and also converges at a faster rate.The Model accuracy Comparison is listed below.

Baseline model[3] 64 Mel spec 3 CNN	Public Map:70 Private Map:69
Our Model	Public Accuracy:64.78 Private :59.7
MinGi Yeon et all Model[5]	Public Map:70.4

Future Work:

In terms of Future work, More works have to be performed on the basis of raw audio than preprocessing.Papers like Wavenet[9] and Attention[8] is all you need have given a nice direction to experiment more with raw audio for discriminative purpose.And also training data should be evaluated by chunking the non verified labels and rank those chunks based on their impact on the model capability to generalize better to new data even with shallow neural networks.This helps to prune the excess noise and to design the better model. If Computation provided, More experiments based on ensembled bidirectional recurrent networks and convolution layers need to be designed so that then model performs much better even for longer audio duration.

Conclusion:

Thus MFCC with Convolution layers helped to classify the audio labels reasonably well even with shallow neural networks. It also helped me to learn about the deep learning model general behaviour to generalization for new data. So with more added noise labels, even the better tuned hyperparameters and optimization technique did not make a big difference and will most likely struck in the local optimum for test data, which can be rectified either with deep neural networks or with pruned training data and better audio preprocessing technique.So in future works,the more emphasis should be placed on input preprocessing and ensemble models for the big improvement in the sound event label classification.

References:

1) https://www.kaggle.com/fizzbuzz/beginner-s-guide-to-audio-data

2) Eduardo Fonseca, Jordi Pons, Xavier Favory, Frederic Font, Dmitry Bogdanov, Andrés Ferraro, Sergio Oramas, Alastair Porter, and Xavier Serra. *Freesound datasets: a platform for the creation of open audio datasets.* In Proceedings of the 18th International Society for Music Information Retrieval Conference (ISMIR 2017), pp 486-493. Suzhou, China, 2017.

3) Eduardo Fonseca, Manoj Plakal, Frederic Font, Daniel P. W. Ellis, Xavier Favory, Jordi Pons, and Xavier Serra. *General-purpose tagging of freesound audio with audioset labels: task description, dataset, and baseline.* Submitted to DCASE2018 Workshop, 2018. URL: https://arxiv.org/abs/1807.09902, arXiv:1807.09902.

4)K. Choi, G. Fazekas, M. Sandler and K. Cho, "Convolutional recurrent neural networks for music classification," 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) , pp. 2392-2396.

5) MinGi Yeom, JaeHoon Moon, “DCASE2018 Challenge: General-Purpose Audio Tagging”

6) https://librosa.github.io/

7) https://en.wikipedia.org/wiki/Mel-frequency_cepstrum#cite_note-1

8) Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin. Attention Is All You Need. URL: https://arxiv.org/abs/1706.03762

9) Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, Koray Kavukcuoglu. WaveNet: A Generative Model for Raw Audio. Submitted on 12 sep 2016. URL: https://arxiv.org/abs/1609.03499v2

10) https://keras.io/