Environmental Sound Classification Using Convolutional Neural Network

Naga Venkata Sai Varanasi, Dept. of Electrical and Computer Engineering, University of Florida, varanasi.n@ufl.edu

***Abstract*—There are many perfect services for speech recognition out there. These are designed to convert speech to text very efficiently but none of them can detect which type of sound it is. Was it glass shattering? Drums rolling? Human voice?**

By creating a model to detect them, these can help to prevent crime in main cities by searching for violent sounds like shooting, human screaming etc. These techniques can also be used to separate speaker from background music or background noise. This system can also be used for people with hearing loss as alert signal for any danger. The concept of audio classification is currently an active research area of research and the best machine learning and deep learning algorithms are still unable to produce reliable results

In this project we will implement an audio classification problem approach using a CNN (Convolutional neural network), analyze its performance and experiment with algorithms which provide performance enhancements. Implement using keras on Urban Sound Classification Dataset.

I. INTRODUCTION

THERE are various number of sounds which ranges from smooth music to heavy violent sounds. Over the past decade advancements in machine learning and AI helped to solve many complicated problems with different kinds of datasets. Sound is one of such data where AI can solve problems like generating various kinds of music, speech-text recognition, speaker recognition and sound classification.

Speech recognition/classification algorithms can date back to early 1960's where Bell Labs designed an algorithm to distinguish between numbers 1 and 9. After that several advancements in the machine learning algorithms and data availability also heavily increased. With the huge data available in present time, neural networks are overthrowing all the other methods in every field including audio classification. There are many types of neural networks available today like ANN, CNN, RNN, LSTM. Each network has its own unique property and advantages. Keras is a tool to easily perform and control deep learning models using python programming language.

Generally, network for working with sound data-set must have a memory capability. This can be easily understood if you think how humans can understand a sentence – They construct and conclude the meaning of every word based on previous words. Some of most popular networks are LSTM and RNN. These networks are designed to recognize patterns, they take the time and sequence into consideration but are more complicated.

Our project shows how sound-classification can be solved through simpler neural network -CNN (Convolutional neural-net) by considering problem as multi-class image classification.
Here the images are spectrogram of the sounds and the classes are various types of sounds.

This project demonstrates how sound classification is achieved efficiently using CNN by performing our experiment on Urban Sound dataset with 10 different classes. CNN analyses the key patterns and determine the class of the data by performing statistical analysis on the spectrogram images.

II. PROJECT DESCRIPTION

A. Urban Sound Classification Database

This project requires a sound dataset with wide variety of classes with significant variance. The Urban sound data set posted in Kaggle is considered for this application [4][5]. This dataset has 8732 sounds for urban sounds with each sound having less than 4 seconds period. It has 10 classes which are Air Conditioner, Car Horn, Children Playing, Dog bark, Drilling Engine, Idling Gun Shot, Jackhammer, Siren, Street Music. All these classes have sufficient amount of variance which can be detected, and data could be classified with the present model.

B. Data Pre-processing

After downloading the data from Kaggle website further preprocessing is done before training our model. The raw data that is downloaded has audios of format .wav and the labels for these audios are defined in .csv file.

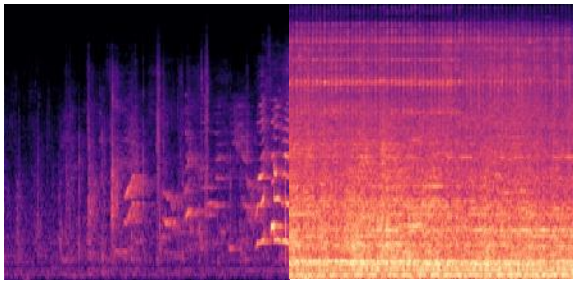
Spectrogram:

Many different types of features could be extracted from the sound data for classification [1]. The CNN model used in this project is designed to work with Image data derived from the music/sound data. Mel-spectrogram is selected as image representation of sound.

Spectrograms are visual representation of sounds which plots between change in loudness(dB) or the frequency over the time period. Unlike a memory-based LSTM network this CNN doesn't recognize the individual words from a sound but can perform pretty good discriminant classification between wide variety of sounds.

We can perform this sound to spectrogram conversion using LibROSA package. Whole set of data is converted into images in batches. We use garbage collector to optimize the memory usage between batch conversions. The output images are stored in separate folder for further processing.

Figure 1: Spectrogram of two different sounds



The above two images are spectrograms of two different sounds. Left part is children playing and right is jackhammer. We can observe there is huge difference in the spectrogram. This is the main motive of selecting CNN for our application.

Data generator and splitting:

Initially out of 8732 images only 5435 are used for CNN network and the remaining data is used for testing. The images that we produced after conversion doesn't have labels embedded to it. Labels are specified in .csv file. We need to embed this information and also we need to split the dataset further info train-validation split.

All these tasks can be achieved using keras one-step datagenerator which takes input images and .csv files as arguments and produces a data representation for keras models based on train-validation split ratio specified. Here in our experiment we use 0.25 as validation split (4077 images for training and 1358 images for validation)

C. CNN Design

The network model designed for this application consists of 6 convolution layers 2 dense full connected layers and pooling-dropout layers in between them. Everything is built on python with keras backend.

Figure 2: Network design

The image shows the order of layers the input is visited before the output layer is reached,

We used 6 convolutional layers with increasing filter density. Initially 32 then 64 and at final stages it reaches 128 filters. This type of design is proved to give much good results historically. This could be understood if you analyses what happens in initial and final layers. Initially only the common features between the images are extracted but at the end the features are so similar that you need high number of filters to differentiate between different images. ReLu is taken as activation function for every convolution layer.

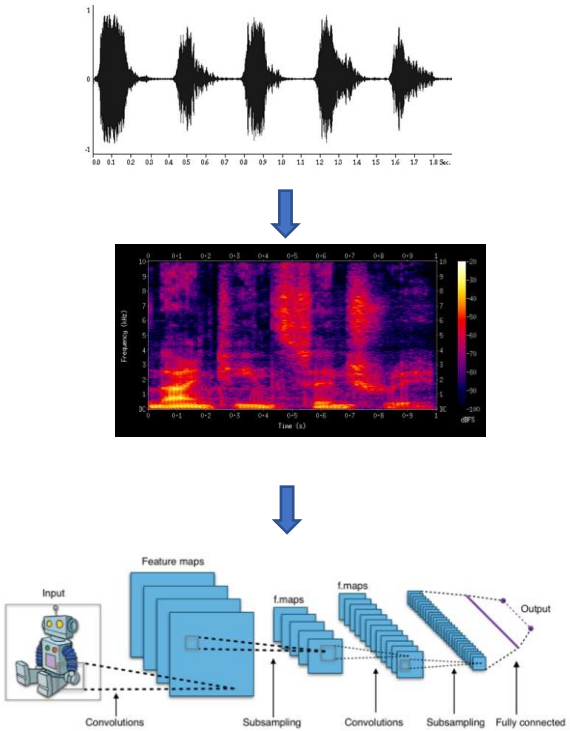
Pooling layers serves the purpose of increasing computation efficiency by down-sampling the data. This also helps in preventing the major problem with excess data in neural networking i.e., overfitting.

Dense layers are placed at the end of the network. These are fully connected layers; the data is generally flattened before the dense layers. We used 2 dense layers in the network. One with 512 neurons and one with 10 neurons. The second dense layer is the output layer with each neuron representing each class of the data.

After the network is designed a optimizer is selected which calculates the error rate and modifies the weights in the network. Gradient descent-based optimizers are used. Out of RMSprop(Root mean square) and ADAM(Adaptive moment estimation) RMSprop is selected as its giving more output accuracy. Learning rate is taken as 0.005. Our network is trained on train part of split and accuracy is evaluated on the validation data of the split.

The project design is represented in the below figure.

Figure 3: Project design



First image is the raw sound data which is converted to spectrogram in second image and is supplied to CNN which is showed in third image.

III. EVALUATION

After the training of our network a number of tests are performed to evaluate the best fit for our data. Several data splits are performed at different stages which are now used to perform the experiments.

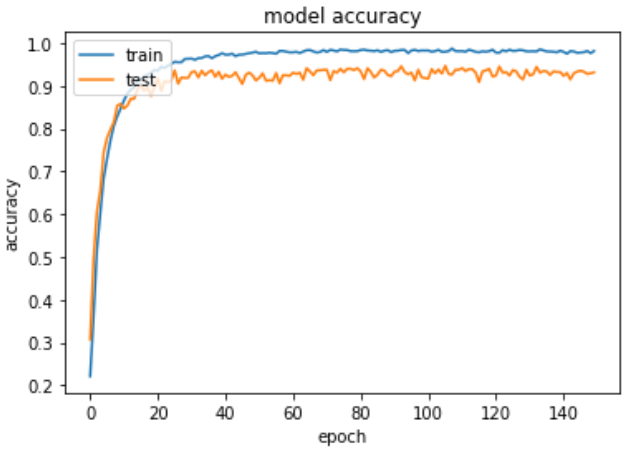
Accuracy during training:

The main step which is performed for every machine learning algorithm/ neural networks is to check accuracy of data during training. Train- validation split of the data will be used for this analysis. The main reason to provide the accuracy test for the data is to change the parameters like learning rate, activation functions, learning heuristics which gives best accuracy. It's not only about highest accuracy, sometimes the neural network has the problem of overfitting the data which results in poor performance when working through new dataset. Accuracy test helps to evaluate this performance.

Out of 5435 images we used 4077 images for training the data and the remaining 1358 are used for validation test. Accuracy is checked after every epoch of training. We have used 150 epochs for training this data.

The result of the performance is displayed in the below figure.

Figure 4: Accuracy on train and validation data vs epoch



Final accuracy of our data on validation test is found to be 95%. Upon further analysis it is found by using less data like 2000 for training the accuracy falls down

to 83%. But if we use more data than present one the network tries to overfit the data learning the whole images and not the particular features which make them unique. Thus Train-validation split is found to be optimum.

Testing on Test dataset:

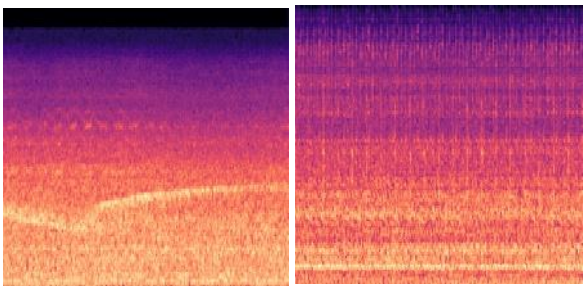
Before training data the data is initially splitted into train and test for future analysis. This data is now sent through the trained network and the outputs are compared with the ground truths available. This step of analysis mainly concentrates on the type of classes and what may be causing errors in the accuracy. Based on the data obtained a confusion matrix is constructed. Below figure shows the confusion matrix.

Figure 5: Confusion Matrix

	AC	Horn	Child	Dog	Drilling	Engine	Gun	Hammer	Siren	Music
AC	203	0	0	0	1	0	0	0	0	5
Horn	0	534	2	0	0	0	0	0	0	0
Child	0	3	232	1	0	0	0	0	19	0
Dog	0	0	0	237	0	0	4	0	0	0
Drilling	1	0	0	0	208	0	0	0	0	1
Engine	2	0	0	0	0	322	0	28	0	0
Gun	0	0	0	4	0	0	186	0	0	0
Hammer	0	0	0	0	0	36	0	364	0	0
Siren	0	4	22	1	0	0	0	0	205	0
Music	0	0	0	2	0	0	0	0	0	592

Even though we are able to achieve 95% of accuracy on the data, the reason behind the error can also be due to type of classes and data we are using. From analyzing the confusion matrix main errors are occurred between the classes of child and siren and also hammer and engine. From further analysis on the spectrograms of the images.

Figure 6: Hammer and Engine sound spectrograms



It could be inferred from the images this error may be due to similarity between images. This may also be due to mis-classification in some of the dataset as the data is labelled by a human and sometimes the sounds look similar which results in human-error in the data-labelling.

IV. BACKGROUND AND RELATED WORK

A huge amount of research work is focused on classification of environmental sounds in past decade. These sounds may be caused by natural or artificial effects such as thunder, gun shot. These sounds help in many different ways wither its from creating the audio annotations for a video or helping elderly person by assisting or alarming.

The main research that happened till now mainly focused on two types of sounds- speech and music. Hidden markov model [8] is one of such approach designed for speech recognition. Unfortunately, this approach doesn't work quite well for the environmental sounds as they are not characterized like a speech, they don't have phonetic model. Considering the research in music recognition, music has a specific pattern [10] which doesn't work for the environmental sounds.

Present work on sound classification is focused on extraction of the features that are relevant and meaningful which distinguishes the variation. The different type of features that could be extracted from sounds are given in this paper [11]. After extracting a certain type of features most of the work related to type of data and classes are done. It's the task of machine learning algorithms to train themselves. Many attempts are done to classify the sound data using SVM [12]. But with the availability in the huge amount of data world is moving towards neural networks. Traditionally sound data is dealt with memory based networks like RNN and LSTM. Recurrent neural network are mainly designed to sequential type of data which may be majorly useful for

speech or music recognition. LSTM (Long-shot term memory) is the advanced version of RNN.

On the other hand major improvements and even more research is going on image data(image processing) CNN is the type of network which is majorly used for these applications. They are much faster than RNN as they don't have memory capability as in RNN and LSTM.

Our idea for this project is the combination of extracting particular features and selecting right neural network for high efficiency and less computational cost. Mel-sprectrogram is one of such features used widely for sound data related experiments. Use of mel-spectrum for voice recognition is demonstrated in this paper[13]. We use this concept to convert our sound data into spectrograms and use those as image data for CNN network designed.

V. SUMMARY AND CONCLUSION

To summarize this project, we showed the application of Convolutional Neural Networks for classification of environmental sound data. We used urban sound data set as a experimental data. Sounds are first converted into images using spectrogram concept and later these images are used for training the CNN. Certain train-test and train-validation splits are performed, and the performance is evaluated. Network is improved based on the performance and the final model gave accuracy of 95%.

REFERENCES

[1] Ghoraani, B.; Krishnan, S.: Time–frequency matrix feature extraction and classification of environmental audio signals. Audio, Speech, and Language Processing, IEEE Transactions on, 19 (7) (2011), 2197–2209.

[2] E. Zwicker, H. Fastl, "Content-based Audio Classification and Retrieval using SVM Learning", IEEE Transactions on Neural Networks, vol.14, no.1, pp 209-215, 2003

[3] J. Salamon and J. P. Bello, "Unsupervised feature learning for urban sound classification," *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brisbane, QLD, 2015, pp. 171-175.

[4] H. Deshpande and R. Singh and U. Nam, "Classification of music signals in the visual domain", Proc. the COSTG6 Conf. on Digital Audio Effects, 2001.

[5] A. Khamparia, D. Gupta, N. G. Nguyen, A. Khanna, B. Pandey and P. Tiwari, "Sound Classification Using Convolutional Neural Network and Tensor Deep Stacking Network," in *IEEE Access*, vol. 7, pp. 7717-7727, 2019.

[6] Comparative Study of CNN and RNN for Natural Language Processing Wenpeng Yin† , Katharina Kann† , Mo Yu‡ and Hinrich Schutze ¨ † †CIS, LMU Munich,

[7] Germany ‡IBM Research, USA {wenpeng,kann}@cis.lmu.de, yum@us.ibm.com. arXiv:1702.01923v1 [cs.CL] 7 Feb 2017

[8] Yu, G.; Slotine, J.-J.: Audio classification from time-frequency texture, in IEEE Int. Conf. on Acoustics, Speech and Signal Processing, 2009 (ICASSP 2009), IEEE, 2009, 1677–1680

[9] Chachada, S., & Kuo, C. (2014). Environmental sound recognition: A survey. *APSIPA Transactions on Signal and Information Processing, 3*, E14. doi:10.1017/ATSIP.2014.12

[10] Rabiner, L.R.: A tutorial on hidden Markov models and selected applications in speech recognition. Proc. IEEE, 77 (2) (1989), 257–286.

[11] Scaringella, N.; Zoia, G.; Mlynek, D.: Automatic genre classification of music content: a survey. IEEE Signal Process. Mag., 23 (2) (2006), 133–141.

[12] Mitrović, D.; Zeppelzauer, M.; Breiteneder, C.: Features for contentbased audio retrieval. Adv. Comput., 78 (2010), 71–150.

[13] SOUND SOURCE CLASSIFICATION USING SUPPORT VECTOR MACHINE Makoto KUMON ∗ Yoshihiro ITO ∗ Toru NAKASHIMA ∗ Tomoko SHIMODA ∗ Mitsuaki ISHITOBI ∗ ∗ Department of Intelligent Mechanical Systems, Graduate School of Science and Technology, Kumamoto University, 2-39-1, Kurokami, Kumamoto, 860-8555, JAPAN

[14] GMS, Corianti & Fahmi, Fahmi & Pinem, Maksum & P Panjaitan, Sihar & Suherman, Suherman. (2018). The use of Neural Network and Mel Frequency Spectrum for voice recognition.

[15] S. E. Kahou, C. Pal, X. Bouthillier, P. Froumenty, C. Gülçehre, R. Memisevic, P. Vincent, A. Courville, Y. Bengio, R. C. Ferrari, et al.
"Combining modality specific deep neural networks for emotion recognition in video.", *Proc. 15th ACM on International conference on multimodal interaction*, pp 543–550. ACM, 2013.