Audio Engineering Society

Convention Paper 10322

Presented at the 148th Convention,2020 June 2-5,Online

A deep learning approach to sound classification for film audio post-production

Guillermo G Peeters[1] and Joshua D Reiss[2]

[1] Queen Mary University of London, Mile End Rd, Bethnal Green, London E1 4NS

[2] Centre for Digital Music, Queen Mary University of London, Mile End Rd, Bethnal Green, London E1 4NS

Correspondence should be addressed to Author (guillem.peeters@gmail.com)

ABSTRACT

Audio post-production for film involves, among other things, the manipulation of large amounts of audio data. There is a clear need for the automation of many organization and classification tasks that are currently performed manually and repeatedly by sound engineers, such as grouping and renaming multiple audio recordings. Here, we present a method to classify such sound files in two categories, ambient recordings and single-source sounds or sound effects. Automating these organization tasks requires a deep learning model capable of answering questions about the nature of each sound recording based on specific stereo and monaural features. This study focuses on identifying these features and on the design of one possible model. The relevant features for this type of audio classification and the model specifications are discussed. In addition, an evaluation of the model is presented, resulting in high accuracy, precision and recall values for audio classification.

1 Introduction

Machine learning methods can successfully classify sound signals using categories such as Noise, Natural Sounds, Artificial Sounds, Speech or Music [1, 2], and across them, i.e. the 'Freesound General-Purpose Audio Tagging Challenge'[1] (2018). The approaches often used objective labels related to the nature of the source [3, 4, 5] (i.e. a recording of a chainsaw might be labelled as man-made since a chainsaw cannot be found in nature) as discretizers.

This project aims to classify sound signals using labels linked to the industry of cinema and video-game audio, such as ambient recordings and textures from sound effects. The boundaries that separate ambient sounds from sound effects are well documented [6]. However, there are no criteria that differentiate these classes in an objective or absolute way and can be generalized to all non-musical and non-speech sound signals.

In film audio post-production, an *ambient sound* is a non-speech, non-musical signal in which one or more components or streams corresponding to different sources, are layered together [6]. A *sound effect* is a non-speech non-musical signal that generally contains a single stream, a sound with a single source.

Previously mentioned models [1-5] are fit for categorization across different types of sound effects and capable of classifying different ambient sounds. However, they are not focused on the discrimination between the two main non-objective classes.

The relevant audio signals do not always easily fit into one of these two classes. More than one stream might be layered in a sound effect to characterise the sound of a single source, i.e. a car sound effect contains the sound of the tires, the motor, etc. or due to practical limitations of the recording process, i.e. the sound effect of a train approaching the station

[1] https://www.kaggle.com/c/freesound-audio-tagging

might have a background of traffic sound or conversations in the station.

This nomenclature, ambient sounds and sound effects, is also not universal. In other contexts, both may be referred to as sound effects, e.g. [7-9]. Furthermore, researchers tend not to use the term sound effects and instead refer to the second category as *single source sounds*. We will generally adopt this term throughout the text, to avoid ambiguity.

Besides, we assume that the original label of the file, given by the engineer who recorded or generated the sound, can be considered as ground truth.

2 Related work

Previous research on audio signal classification often made use of frequency-based features for sound classification [2, 10]. Either single source sounds or ambients might produce a large variety of spectrograms since both might have any possible source (or combination of sources). Therefore, features such as the Mel Frequency Cepstral Coefficients (MFCCs) and Mel Spectrograms, which can be treated as images that capture the frequency characteristics of the sound signal, result to be insufficient and misleading for this research [11, 12].

The class definition conflicts explained in the previous section, reside in the frequency domain, that is, Mel Spectrograms might help differentiate the sound of two different types of vehicle and however might not succeed in categorising if a sound file is a single source sound or an ambience recording.

Thus, the features to be extracted for the building of a Machine Learning model capable of this sort of classification have to be picked-by-hand out of different libraries and previous studies, as opposed to the use of the available feature libraries, which are of valuable and common use in other types of audio classification problems.

Fortunately, a lot of previous work in audio signal classification is still a good reference because it makes use of complementary features that, although they may not solve classification problems as standalone features, can be of great importance in this study, i.e. the use of Spectral Bandwidth (SB) feature as a complement of the Spectral Centroid (SC).

In addition, previous research not directly related to this study has been revised and used in this classifier [13]. Stereo width features have been extracted from Brecht De Man et al. [14], an analysis of valuable features for multitrack music mixtures. Energy-Entropy (EE) features, often used in speech detection problems [15], are adapted to help classify our signals.

Since entropy is a measure of the uncertainty and disorganisation of a random variable, it can be a good indicator of the degree of randomness of an ambient recording in comparison to less random single source sounds such as alarms and tone like sounds.

Other more common features were also adapted and implemented in this model. Zero-Crossing Rate (ZCR) [16] measures the number of times in a given time interval that the amplitude of the signal crosses through a 0 value. Ambient recordings and sound textures are expected to have a fairly constant distribution of frame zero-crossing rates, as opposed to single source sounds which will contain periods with low ZCRs dispersal and periods with high ZCRs dispersal.

3 Implementation

3.1 Data

The data used for testing and training the model consisted of two sets of audio samples commonly used in movie sound postproduction and the original audio files of a real feature film.

The BBC Sound Library[2] includes 16,000 sound files available for personal, education or research purposes. The 6000 Sound Effect bank[3] includes 7,500 sound files containing single source sounds and ambient recordings.

The original film audio files contain professional sound from the set recordings of the unreleased movie ‘Emperor (2014)’ and around 1,643 Foley effects. This material was facilitated by Nick Lowe, a professional sound designer and sound supervisor who has worked on more than 64 titles.

[2] https://bbcsfx.acropolis.org.uk/

[3] https://www.sound-ideas.com

The data types range between mono, stereo and multi-channel wave audio files of different lengths. The original sampling frequency of the files is 48kHz and they are quantized at 16 or 24 bits (the 6000 Sound Effect bank was recorded for CD format). All the files are either single-source sounds or ambient sounds including indoors and outdoors recordings.

The data preparation and feature extraction processes have been implemented in Python[4] using the Librosa[5] and pyAudioanalysis[6] libraries.

All files were trimmed to the length of one minute, files shorter than one minute are lengthened by zero-padding. The trimming function is applied with a one-second offset to avoid short silences at the beginning of the audio files.

44.1kHz was chosen as sampling frequency since preserving the high-frequency content of the signals is key for a correct classification of the audio files.

Only the first two channels of the multi-channel files were preserved, taking into account that in the field of cinema, by convention, the first two channels of a multi-channel file belong to the 1st and 2nd boom microphones and the remaining channels belong to radio microphones (lavalier microphones).

L1 normalization is applied once stereo features are extracted and the files converted down to mono, to avoid compromising the amplitude differences between L and R channels in the case of one-sided panned stereo single source sounds or ambiences.

As mentioned, the ground truth labels (single source sound or sound texture) were extracted based on the judgement of the sound engineer responsible for the recordings (i.e. the file name or folder origin).

3.2 Stereo features

Two stereo measures from studies on music mix engineering [14] were implemented.

3.2.1 Left/Right Imbalance

The Left/right imbalance is obtained by calculating the Root-Mean-Square (RMS) value of each frame of each channel of the time-series signal. In the L/R imbalance ratio, denoted *L* is the total RMS power of the left channel, denoted *R* the total RMS results of the right channel [14].

3.2.2 Side Mid Ratio

The side/mid ratio measure is the relation between the power of the side-channel and the power of the mid-channel. The side-channel is the addition of the left channel and polarity-reversed right channel divided by the number of channels. The mid-channel is obtained by calculating the RMS power of the addition of the left and right signals divided by the number of channels.

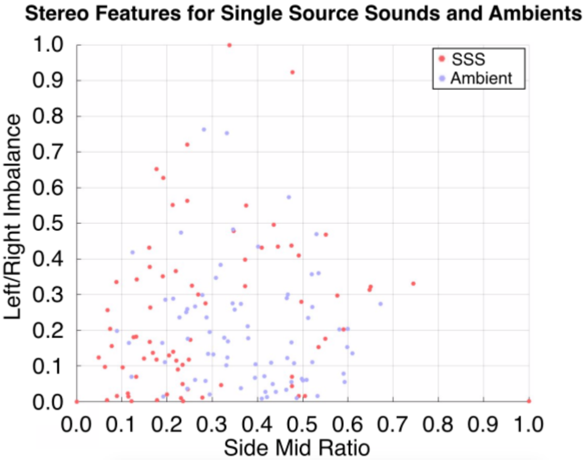

Figure 1. Stereo features (SMR and LRI) for single-source (SSS) and ambient sounds scatter plot.

Taking these two features into consideration, information about the amount of panning of a stereo recording is obtained [14]. We can also extract information about how correlated are the Left and Right signals in each file. For example, the case of a recording using two microphones set in the same exterior location but separated by a considerable distance indicates a low imbalance and a high side/mid ratio which gives low values for correlation.

Figure 1 shows the distribution of single-source sounds and ambient recordings based on their stereo features.

[4] https://www.python.org/

[5] https://librosa.github.io/librosa/

[6] https://github.com/tyiannak/pyAudioAnalysis

3.3 Mono Features

The following mono features were implemented.

3.3.1 Variance of Zero-Crossing Rate (ZCR)

The zero-crossing value extracted is an average of the variance of the rate of zero-crossings over the total number of frames of the audio file. Here, the ZCR [2] was not used as a pitch tracker as usual, since previous work showed that ZCR is informative as a stand-alone feature [17, 18].

Ambient recordings and textures have a normal distribution of frames with lower and higher ZCRs. In contrast, single-source sounds contain a much more uneven distribution; distinct periods with a low number of zero-crossings and periods with a higher number of zero-crossings [2].

The variance of ZCR is extracted from the audio files before zero-padding to avoid the bias of analysing fixed-length files.

The Standard Deviation (SD) measure resulted in high deviation values for single-source sounds and low variance values for constant ambient recordings and textures.

3.3.2 Root Mean-Square Rate (RMS)

RMS [14] levels are obtained by calculating the root-mean-square value of the amplitude of each frame. Although ambient recordings are often used as a background in sound postproduction, i.e. mixed at low amplitudes, they show higher RMS values than single-source sounds, the reason for this is the lack of discernible peaks on the signal. The low dynamic range of ambient recordings allows them to achieve large amplitudes after sound normalization. In contrast, single-source sounds, with higher dynamic ranges, are not affected as much by normalization since their peaks reach clipping levels faster (i.e. after less amplification).

3.3.3 Spectral Features

The spectral features used for this model are the Spectral Centroid (SC), the Spectral Bandwidth (SB), the Spectral Flatness (SF) and Spectral Roll-Off [19, 21, 22].

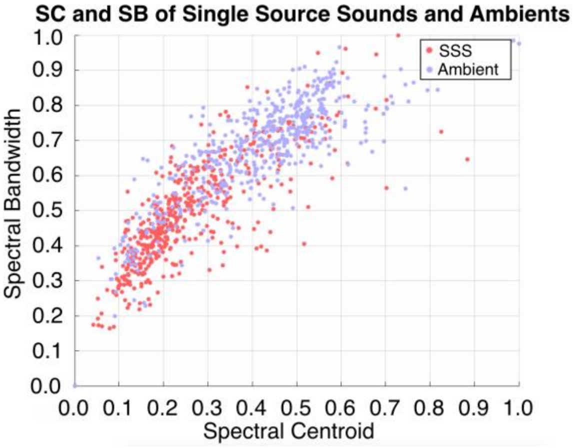

Figure 2. SC (x-axis) and SB (y-axis) values of single-source sounds and ambient sounds.

Although either single source sounds or ambient recordings may show a large diversity of spectral centroids, ambient recordings have a richer high-frequency content, which results in higher values for SC.

In addition, ambient recordings also present higher values for Spectral Bandwidth in comparison to single-source sounds. The Spectral Bandwidth (SB) or Spread is calculated by taking the frequency difference of each spectrum in relation to the Spectral Centroid (SC) [19].

The bandwidth of the spectrum is valuable information for the reason that single source sounds are often recordings of a single source, consequent repetitions of the same sound recorded usually in a silent environment, in contrast, ambient recordings contain much more components. This fact leads to lower values of Spectral Bandwidth for single-source sounds and higher values for ambient recordings or textures (figure 2).

Spectral Flatness [19] is a measure of the noisiness of the spectrum. The definition of SF as a measure of noisiness has its origins in MIR research [20]. In equation 1, SF is calculated by taking the geometric mean of the spectrum and dividing the values by its arithmetic mean.

$$SF(m) = \frac{(\prod_k |X(m,k)|)^{\frac{1}{k}}}{\frac{1}{k}\sum_k |X(m,k)|} \quad (1)$$

Where k is the band number and K the number of frequency bands and k band number.

In the case of this study, the data (non-speech non-musical signals) are rarely tone-like, however, the amount of noisiness is higher for ambient recordings than for single-source sounds resulting in higher Spectral Flatness values.

The Spectral Roll-off (0.9) feature, measures the frequency below which most of the spectral content of a signal is located. A 0.9 roll-off value indicates that 90% of the spectral energy is located below the measured frequency (i.e. this parameter could be set to 1.0 to find the highest frequency in the spectrum). This feature was originally used as a transient detection method in MIR [21] and it is now popularly used as a discretizer between voiced and unvoiced speech [22]. In our context, it is another spectral measure of the high-frequency content of the data.

3.3.4 Energy (E) and Energy Entropy (EE)

This feature calculates the Energy of each frame of a signal windowed using a rectangular window. Various studies use entropy-based algorithms in the field of speech detection [13] [23]. Entropy is defined as the amount of uncertainty in a random variable. Energy Entropy is implemented gathering that single source sounds will have more organized segments than noisy ambient recordings, resulting in lower entropy values. Energy Entropy is defined by equation 3.

$$E = \frac{x(n)^2}{W} \quad (2)$$

$$EE = -\sum_i (s_i \cdot \log_2(s_i)) \quad (3)$$

Where $s_{(i)}$ are each of the sub-frame energies normalized by dividing by the total energy (E) of each frame. The sub-frame energies are obtained by using equation (2), after a secondary window with shorter width is applied.

3.3.5 Mean Onset Strength (OS)

The onset strength of a time-domain signal is obtained by calculating the spectral flux onset strength envelope. Onsets are the amplitude peaks in a signal over a defined threshold. The strength of each onset is measured by taking two consecutive short-time spectra and calculating the energy difference between them, bin by bin. Following that, each non-negative and non-zero difference is added together [24].

The audio signal is divided into overlapping frames of 2048 samples, subsequently, the frames are windowed by a Hann window of the same sample length. The signal is transformed to the frequency domain using the Discrete Fourier transform (DFT) and the spectra are obtained by taking the Log-power Mel Spectrogram of the signal using, by default, 256 Mel Frequency Bands. The frame reference is a result of the local maxima filtering along the frequency axis. The results of these calculations per each frame take the shape of an onset envelope describing the signal's amplitude changes over time. The mean strength of all the onsets is calculated and added to the feature data.

Single source sounds have higher strength onsets since these are recordings of higher amplitude sounds i.e., impacts, gunshots. The information provided by this measure differs from the RMS feature for the reason that only the amplitude of the onsets is calculated and the amplitude of the inter-onset frames is ignored.

3.3.6 Number of peaks

The number of detected peaks in each onset strength envelope is counted. The parameters of the peak picking function set the rules that define which onsets are to be declared peaks.

In this model, a peak is detected under the following two conditions:

If a sample of the onset envelope (*x[n]*) has a higher amplitude than its previous 3 samples (*x[n-3]*) and its respective 3 consecutive samples (*x[n+3]*), i.e., every sample of higher amplitude than its surrounding 6 samples is listed as a peak.

If a sample has higher or equal amplitude than an amplitude reference value. Where the reference amplitude is the mean of its previous three samples (*x[n-3]*), its consequent 5 samples (*x[n+5]*) and the sample in question (*x[n]*) added to a threshold value of 0.5 amplitude.

With these parameters, it is possible to detect peaks in ambient sound files despite its immanent continuous characteristics. In contrast, slightly fewer peaks are detected in single-source sounds.

The number of peaks is averaged over the length of the files to avoid longer signals biasing the results due to having more peaks.

3.4 Classification Model

A Neural Network was used for classification. The algorithm decides the most accurate combination of parameters to increase classification accuracy when categorising the test data. After the model is trained, the unseen prediction dataset is given as the input to the model, now capable of classifying each of its rows by using labels.

Previous work on automated sound classification often implements Convolutional Neural Networks (CNNs) and categorical cross-entropy methods for multi-class classification [25], Fully Connected Neural Networks such as 3-layer feed-forward structures are also effectively used for audio clustering by using radial basis functions and K-Means classification [26].

A Fully Connected Neural Network has a simpler architecture that better fits the binary classification problem proposed in this study and was therefore implemented.

3.4.1 Input Layer

For the model to work correctly, each column of the input data matrix, representing each feature, must be on a similar scale. To accomplish this, a standardization function [27] was applied to the feature vectors. This function removes the mean of the data and scales it to unit variance (SD=1).

3.4.2 Hidden Layers

The first hidden dense layer contains 128 neurons and the second layer 64 neurons. The weights in this model are initialized using the Glorot Uniform initializer developed by Xavier Glorot [28]. The activation functions used in this model are Rectified Linear Activation Units (ReLU)[7].

3.4.3 Output Layers

The output layer's activation function is the Softmax Activation[8][29], capable of outputting a tensor with a softmax classification. Softmax classification calculates the probabilities of a single event over the rest of possible events causing all the probabilities in the sample space to add to 1.

3.4.4 Optimization

The predictions obtained during the forward-pass are compared to the actual classes using a loss function. The log-loss is calculated by the binary cross-entropy function. It outputs higher loss values as the predicted probability diverges from the actual label.

During backward propagation of errors, the weights are updated using the stochastic gradient descent algorithm (SGD), which iteratively optimizes the error by finding its minimum.

3.4.5 Regularization

A Dropout (0.2) layer is located between the two hidden layers. Dropout layers are regularization methods, used to stop the model from overfitting.

The probability dropouts caused by this layer, result in alterations in the architecture of the network due to the changes in the responsibilities of all the 'non-dropped-out' neurons, which are forced to maintain the unit probability space. In our case, the probability of 2 out of 10 inputs will be set to 0 (probability rate = 0.2).

[7] https://arxiv.org/abs/1803.08375.pdf

[8] https://github.com/fchollet/keras

	Act. Shape	Act. Size	# Parameters
Input	(12, 1)	12	0
Dense *(relu)*	(None, 128)	128	1664
Dropout *(0.2)*	(None, 128)	-	0
Dense *(relu)*	(None, 64)	64	16512
Dense *(softmax)*	(None, 2)	2	258

Table 1. The architecture of the model: Layers, activation outputs, size of the activations and number of learnable parameters.

4 Evaluation

Different variations of the model, related to the training data balance, features and parametrization (such as learning rate, number of layers, regularization) have been tested during the design process of the classifier. The results and performance of the model were compared using common evaluation metrics in the field of audio classification.

4.1 Evaluation Metrics

The classifier splits the data into two classes, single-source sounds (label 0) and ambient sounds (label 1). Thus, *true negatives* are all the correctly predicted single sound sounds, and *true positives* are the correctly predicted ambient sounds. The term *False positives* refers to the predicted ambient recordings that are actually labelled as single source sounds and *false negatives* refers to the predicted single sound sounds labelled as ambient recordings.

Accuracy refers to the percentage of correct answers. Namely, the ratio of true positives and true negatives over all the predicted data.

Recall or *True Positive Rate (TPR)* refers to the sensitivity of the model, the number of correctly identified ambient recordings (TP) over the total actual ambient recordings in the dataset, it is defined by the expression, *TPR = TP / (TP + FN).*

Specificity or *True Negative Rate (TNR)* defines the number of correctly predicted single sound sounds (TN) over the number of actual single sound sounds files: *TNR = TN / (FP + TN).*

The *Precision* of the model is the number of correctly identified ambient recordings (TP) over the number of identified ambient recordings, it is calculated by *TP / (TP + FP).*

The *False Positive Rate (FPR)* or *Type I Error* is the number of incorrectly predicted ambient recordings (FP) over the total number of actual single sound sounds, *FP / (FP + TN).*

Finally, the *False Negative Rate (FNR)* or *Type II Error* refers to the number of wrongly identified single sound sounds (FN) over the total number of actual true ambient recordings, *FNR = FN / (FN+TP).*

4.2 Results

The results shown in this section have been replicated by shuffling the data, training the model and making new predictions from scratch. The optimisation for the gradient descent is set to a learning rate of 0.01 and the batch size of the model is set to 32. The length of the training data was 1000 samples (audio files) divided into 500 single source sounds and 500 ambient recordings. Using a batch of 32 samples means dividing the dataset into 31 batches; as a result, each epoch will update the weights of the model 31 times. Table 2 shows the results where 500 epochs were set for the training process.

	Precision	Recall	f1-score	Support
SSS	0.83	0.89	0.86	175
Ambient	0.86	0.80	0.83	158
Micro average	0.85	0.85	0.85	333
Macro average	0.85	0.84	0.85	333
Weighted average	0.85	0.85	0.85	333
Confusion	Matrix			
155	20			
31	127			

Table 2. Precision, recall, f1-Score and Confusion Matrix for Model 1.

The accuracy of this model is 84.68%: 282 files were correctly found amongst all the predicted files (333 files). These 333 files are divided into 158 ambient recordings and 175 single source sounds.

The F1-score and the recall measure were slightly higher for single-source sounds, whereas ambient recordings showed higher values for the precision measure.

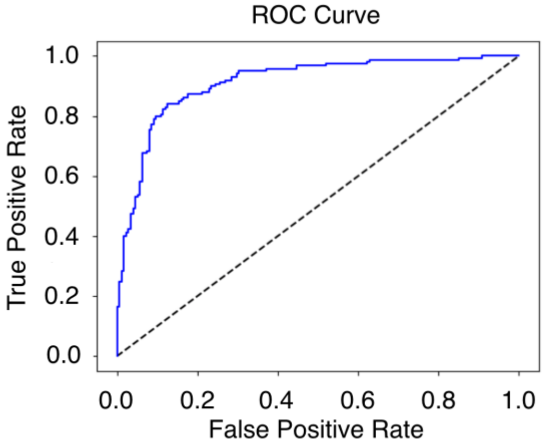

Figure 3. ROC AUC curve showing the performance of the classifier.

ROC AUC curves, shown in figure 3, represent the performance curve of the model at different probability thresholds. This does not take into consideration the prior of probabilities inherent on an imbalanced dataset. However, in this analysis there is only imbalance of prediction data. The train and test data used are exactly balanced about classes. The ROC AUC value is 0.912, the size of the area below the blue line represents how well the model can separate the two classes.

Simpler network structures were attempted by deleting hidden layers from the network and by deleting the less important features. In addition, attempts to avoid overfitting, such as adding more dropout layers, decreasing the gradient descent learning rate and decreasing the Batch Size did not improve the performance of the model in any aspect.

5 Discussion

Though the number of epochs chosen for the Model was 500, each training process was first attempted using the early-stop technique. This method stops the process once the decrease of the loss values, calculated by the loss function, becomes stable. It avoids overtraining since it stops the model once it stops learning. The results using early-stop did not alter the performance of the model and, therefore, the default number of epochs is set to 500.

Starting from a simple model with a structure of only 1 hidden dense layer, formed by 24 nodes, and 1 dense output layer of 2 nodes, the classifier was gradually improved by adding deeper layers capable of better understanding the data and improving its performance. These intermediate models have not been described in the evaluation section since they are all considered drafts of the presented model.

Figure 4, shows the PRC (Prediction-Recall curve) of the Model. It gives a more accurate insight on the performance of the model, taking into account that there is a slight class imbalance of prediction data. Namely, PRC plots compensate for the previous probabilities of each class by not taking the True Negative Rate (or specificity) into account. However, both PRC and ROC graphs show a similar response of the Model. Thus, the imbalance of the prediction data is not biasing the results of the classifier.

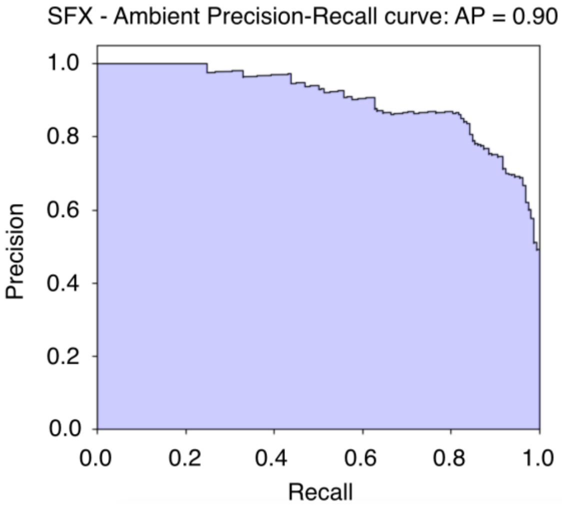

Figure 4. Prediction Recall curve of the Single Source Sound/Ambient Sound classifier.

6 Summary and further work

We proposed a model for the categorisation of audio files into two classes, ambient recordings and single-source sounds, based on analysis of signal features. The features were picked by hand, from previous research in sound classification. Some of the features performed successfully without further adaptation, though many were adapted since they had not been implemented before in the context of this type of non-objective class categorization.

A Fully Connected Neural Network capable of interpreting the features was designed. It departed from a simple structure by adding more layers and parameters to its architecture without losing sight of different evaluation metrics. Once the performance of the model stopped to improve in relation to its complexity, other techniques such as regularisation and elimination of features were explored.

The resulting model outperformed the rest of implementations on the classification task with an accuracy of 84.68% and an f1 score of 0.83. These evaluations are positive and the results should be replicated using larger datasets.

The recording techniques of each sound engineer and the type of microphones used, affect the characteristics of the recordings. So, though the training, test and prediction materials were extracted from three different sources, more evaluations of the model should be done using data from other films and different sound banks.

Further work could be done in exploring the categorisation of other types of cinema sound files such as categorising the audio files by location or in the case of speech audio files by speaker (or actor). Together, these classifiers could result in a more intelligent sound classifier capable of organizing all the pre-processed sound material of a feature film or videogame.

Another application to be explored is using the model for sound segmentation. This would help extract ambient tracks out of long clips recorded on a film, i.e. segments within the moments of speech in the recording of a scene. These extracts could be useful tools for solving continuity problems caused by the image editing of a film.

References

[1] G. Tzanetakis and P. Cook, ‘Musical genre classification of audio signals,’ in *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293-302, July 2002. doi: 10.11097/TSA.2002.800560.

[2] J. P. Woodard, ‘Modeling and classification of natural sounds by product code hidden Markov models,’ in *IEEE Trans. Signal Processing*, vol. 40, no. 7, pp. 1833-1835, July 1992.

[3] Moffat, DJ; Ronan, D; Reiss, JD ‘Unsupervised Taxonomy of Sound Effects’, 20th International Conference on Digital Audio Effects (DAFx-17), September 2017.

[4] Ian Stevenson, “Soundscape analysis for effective sound design in commercial environments,” in Sonic Environments Australasian Computer Music Conference. Australasian Computer Music Association, 2016

[5] PerMagnus Lindborg, “A taxonomy of sound sources in restaurants,” Applied Acoustics, vol. 110, pp. 297–310, 2016.

[6] D. Bordwell and K. Thompson, Film Art: An Introduction, McGraw-Hill, New York, 5th edition, 1997.

[7] P. Bahadoran, et al., "FXive: investigation and implementation of a sound effect synthesis service," International Broadcasting Convention (IBC), Amsterdam, 2018.

[8] D. Moffat, D. Ronan and J. D. Reiss, 'Unsupervised taxonomy of sound effects,' 20th International Conference on Digital Audio Effects (DAFx-17), Edinburgh, UK, September 5–9, 2017

[9] D. Moffat and J. D. Reiss, 'Perceptual Evaluation of Synthesized Sound Effects,' ACM Transactions on Applied Perception, 15 (2), April 2018

[10] Yu, Guoshen and J.-J. E. Slotine. ‘Audio classification from time-frequency texture.’ *2009 IEEE International Conference on Acoustics, Speech and Signal Processing* (2008): 1677-1680.

[11] Edgar Hemery and Jean-Julien Aucouturier, “One hundred ways to process time, frequency, rate and scale in the central auditory system: a pattern-recognition meta-analysis,” Frontiers in computational neuroscience, vol. 9, 2015.

[12] Lagrange, M., Lafay, G., Defreville, B., and Aucouturier, J. J. (2015). The bag-of-frames approach: a not so sufficient model for urban soundscapes, after all. *J. Acoust. Soc. Am. Express Lett.* (accepted).

[13] R. Hariharan, J. Hakkinen and K. Laurila, ‘Robust end-of-utterance detection for real-time speech recognition applications’, *IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (ICASSP)*, Salt Lake City, UT, USA, 2001, pp. 249-252 v.1. doi: 10.1109/ICASSP.2001.940814

[14] B. De Man et al., ‘An analysis and evaluation of audio features for multitrack music mixtures’ 15th Intl. Society for Music Information Retrieval Conference (ISMIR), Taipei, Taiwan, 2014.

[15] T. Kristjansson, S. Deligne, P. Olsen. Voicing Features for Robust Speech Detection. In *6th Interspeech 2005 and 9th European Conference on Speech Communication and Technology (Eurospeech)*, pp. 369-372, 2005.

[16] C. Panagiotakis and G. Tziritas, ‘A Speech/Music Discriminator Based on RMS and Zero-Crossing,’ IEEE Trans. Multimedia, v. 7 (1), p. 155-166, Feb. 2005.

[17] D. S. Shete et al. Zero-crossing rate and Energy of the Speech Signal of Devanagari, 2014, IOSR Journal of VLSI and Signal Processing (IOSR-JVSP) v. 4 (1), Ver. I PP 01-05 e-ISSN: 2319 – 4200, p-ISSN No.2319-4197, Jan. 2014

[18] E. Scheirer and M. Slaney. Construction and evaluation of a robust multifeature speech/music discriminator. In International Conference on Acoustics, Speech and Signal Processing, volume II, pp. 1331–1334. IEEE, 1997.

[19] S. Dubnov, Generalization of Spectral Flatness Measure for Non-Gaussian Linear Processes. *IEEE Signal Processing Letters*, 2004*. 11. 698 - 701. 10.1109/LSP.2004.831663*

[20] A. Klapuri A, M. Davy, Signal processing methods for music transcription. Springer, New York, 2006.

[21] P. Leveau et al. On the usefulness of differentiated transient/steady-state processing in machine recognition of musical instruments. ENST; Laboratoire d’Acoustique Musicale. *In AES 118th Convention, Barcelona, Spain, 2005.*

[22] M. Kos et al., ‘Acoustic classification and segmentation using modified spectral roll-off and variance-based features’, Digital Signal Processing v. 23 (2), pp. 659-674, March 2013.

[23] W.S. Ching, P.S. Toh. Enhancement of Speech Signal Corrupted by High Acoustic Noise. *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '79.*

[24] S, Böck and G, Widmer, ‘Maximum filter vibrato suppression for onset detection’, Proc. of the 16th Int. Conference on Digital Audio Effects (DAFx), Maynooth, Ireland, Sept. 2-6, 2013

[25] Becker, Sören et al. “Interpreting and Explaining Deep Neural Networks for Classification of Audio Signals.” *ArXiv* abs/1807.03418 (2018)

[26] Dhanalakshmi, P., S. Palanivel, and Vennila Ramalingam. ‘Classification of audio signals using SVM and RBFNN.’ *Expert systems with applications* 36.3 (2009): 6069-6075.

[27] F. Pedregosa *et al*., ‘Scikit-learn: Machine Learning in Python’, Journal of Machine Learning Research 12 pp. 2825-2830, 2011.

[28] X. Glorot, Y. Bengio ‘Understanding the difficulty of training deep feedforward neural networks’, Intl. Conference on Artificial Intelligence and Statistics (AISTATS), 2010.

[29] I. Goodfellow, Y. Bengio and A. Courville, ‘Deep learning’. The MIT Press, 2016, 800pp, ISBN: 0262035618. doi: 10.1007/s10710-017-9314-z