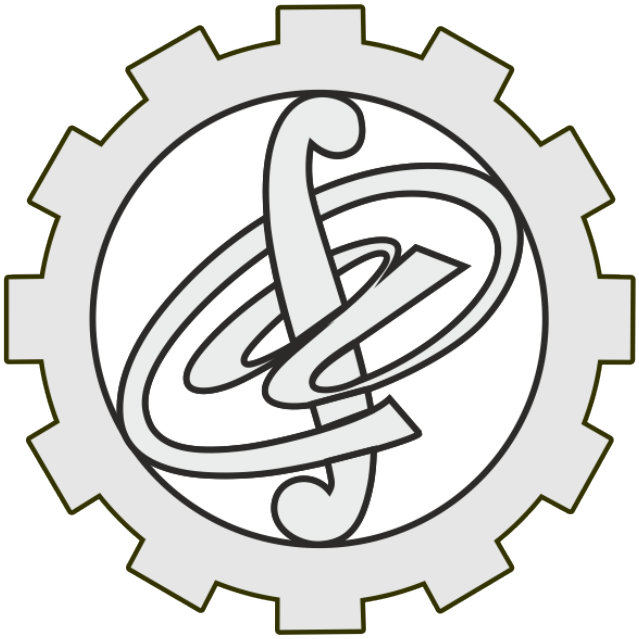
Deep Learning in theory and applications

Project report - Comparison of different approaches to urban sound classification

Patryk Gamrat, Karol Zając
Silesian University of Technology, Faculty of Applied Mathematics

January 27 2025



Contents

1 Introduction

1.1 Problem

With the growth of machine learning technology, it has become possible to automate various complex processes, including monitoring our environment for potential dangers. This advancement allows us to enhance safety and respond to hazardous situations more quickly and reliably. In this project, we aimed to develop a system trained to identify environmental sounds based on their Mel spectrograms, enabling automated detection of potentially dangerous situations such as gun violence, dangerous animals, or traffic accidents.

1.2 Target groups

Our deep learning network could be useful for many professions, here are some examples:

- **First responders** - A system using the network could notify the appropriate services of a potentially dangerous situation as soon as it is detected. This saves time and allows for faster reactions, which is crucial when citizens' health or lives are in danger. For instance, when gunshots are detected, police would be alerted, the system could help reroute traffic when ambulance sirens are detected.
- **Urban planners** - Data gathered from around a city and analyzed by the network could help recognize problems with the infrastructure. For example, it could identify where most accidents occur or which places experience a large amount of traffic, aiding in better urban planning. Additionally, the system could be integrated with navigation apps to advise drivers to avoid areas with too much traffic.
- **Citizens aiming to protect their homes** - The system could analyze audio data gathered from around a property for any signs of danger, helping prevent robberies or acts of vandalism. The system could then alert the property owner, security or the police. For example, we could listen for engine idling outside home or other suspicious sounds and alert owner via mobile app
- **Noise pollution reduction** - big cities often face problems with noise pollution that affects peoples health and well-being, noise classification system could identify source of noise in busy city areas to help authorities address this problem

1.3 Other potential applications

While in this project we focused on classifying urban sounds, with further development and training on new data, our approach could lend itself to a plethora of interesting applications.

- **Industrial machinery monitoring** - A system to analyze the noise produced by machinery to detect possible malfunctions early, increasing safety and lowering maintenance costs.
- **Nature reserve protection** - This approach would allow monitoring for possible illegal human activity like poaching. Additionally, it could help study animal behaviour by detecting various animal sounds.

- **Assistive technologies** - Sound detection could help those with impaired hearing by providing information about their surroundings, increasing safety.

2 Model

2.1 Theory behind audio classification systems

Spectograms and Mel frequency coeeficients (mfcc's) can be used to classify audio. Both of these features derive from the mel scale, which has properties that closely models human perception of sound and difference in pitch. Extracting these features requires a few steps, that mostly make use of the Short-time Fourier transform (STFT).

We tried two approaches to classifing urban noises. First approach uses mel frequency coeeficients (mfcc's) that are used as features in a multilayer perceptron neural network. The second approach uses mel spectogram images as direct input to a convolutional neural network, to classify sounds into one of 10 classes.

2.2 UrbanSound8k dataset

UrbanSound8k dataset used to train our models consists of more then 8000 audio samples split into 10 classes that represent common city noises.

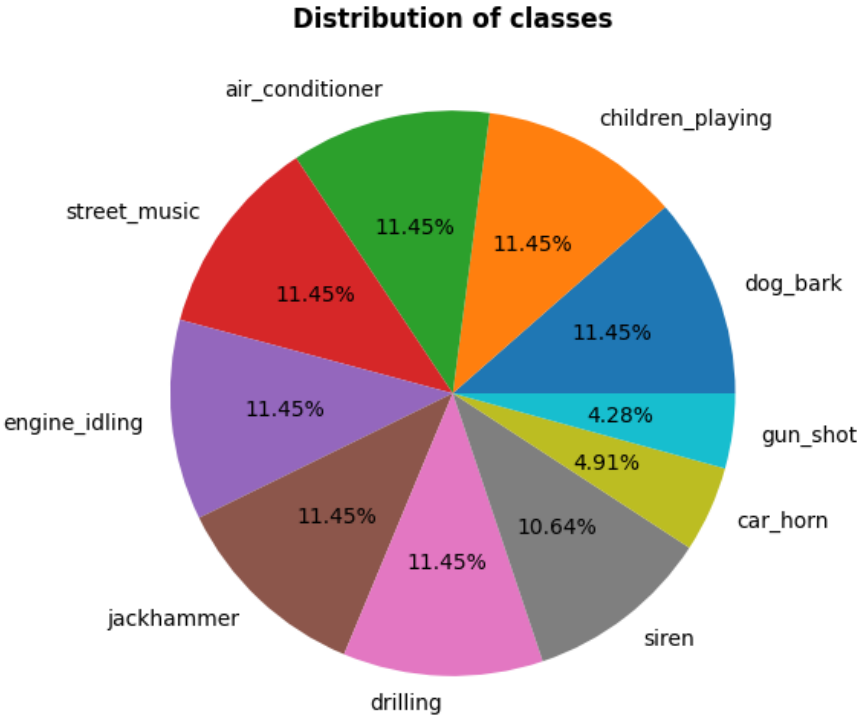


Figure 1: Distribution of classes in UrbanSound8K dataset

Audio samples in our dataset are split into foreground and background noises, with most samples being foreground. For background noises the target is being distrubed by other surrounding noises. For this reason, classifying background noise might be more challenging.

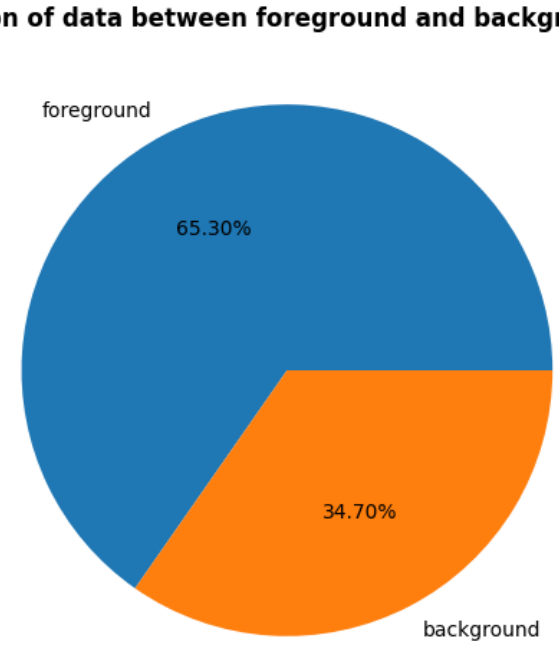


Figure 2: Disribution between foreground and background noise

We can visualize classes using PCA Decomposition.

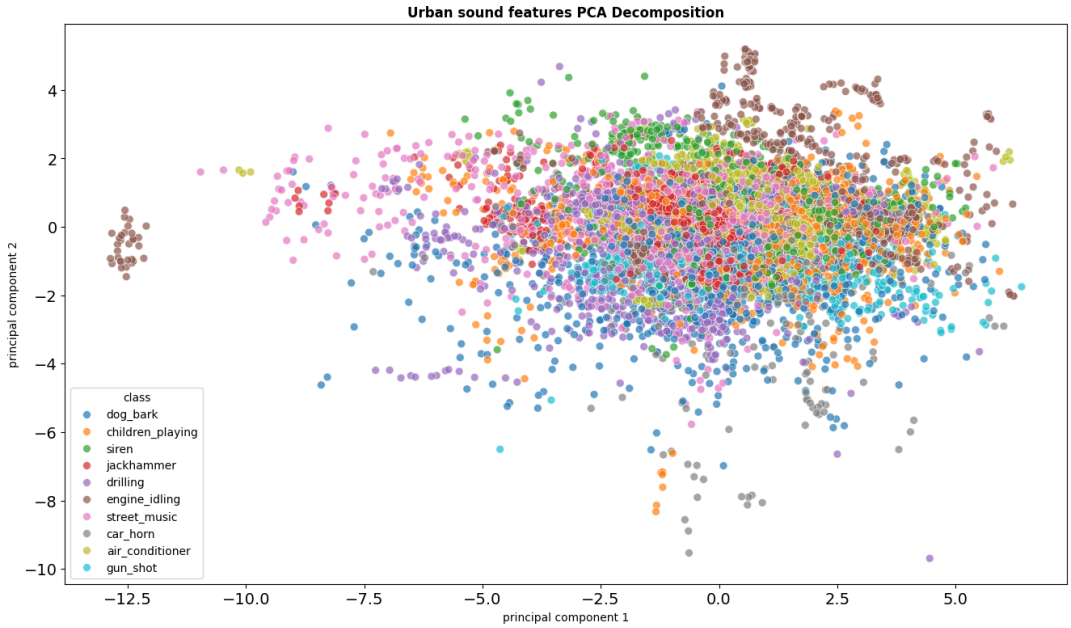


Figure 3: PCA visualization of sound classes

2.3 Approach 1 - Multilayer Perceptron Network

Our first approach to this problem uses mel frequency cepstrum coefficients to classify audio samples. Since for each coefficient we get results in time series format, as a feature we use the mean of the values over time. This is also necessary, because our audio samples are varying in length. For our proposed solution we extracted 25 coefficients, since extracting more doesn't provide any measurable benefits in terms of performance.

Data was split into 80-10-10 split (80% for training, 10% for validation and 10% for final testing). Standard scaler was used to normalize the data.

We can visualize the correlation of features on a correlation matrix. Some correlation can be observed which is normal for mffc features.

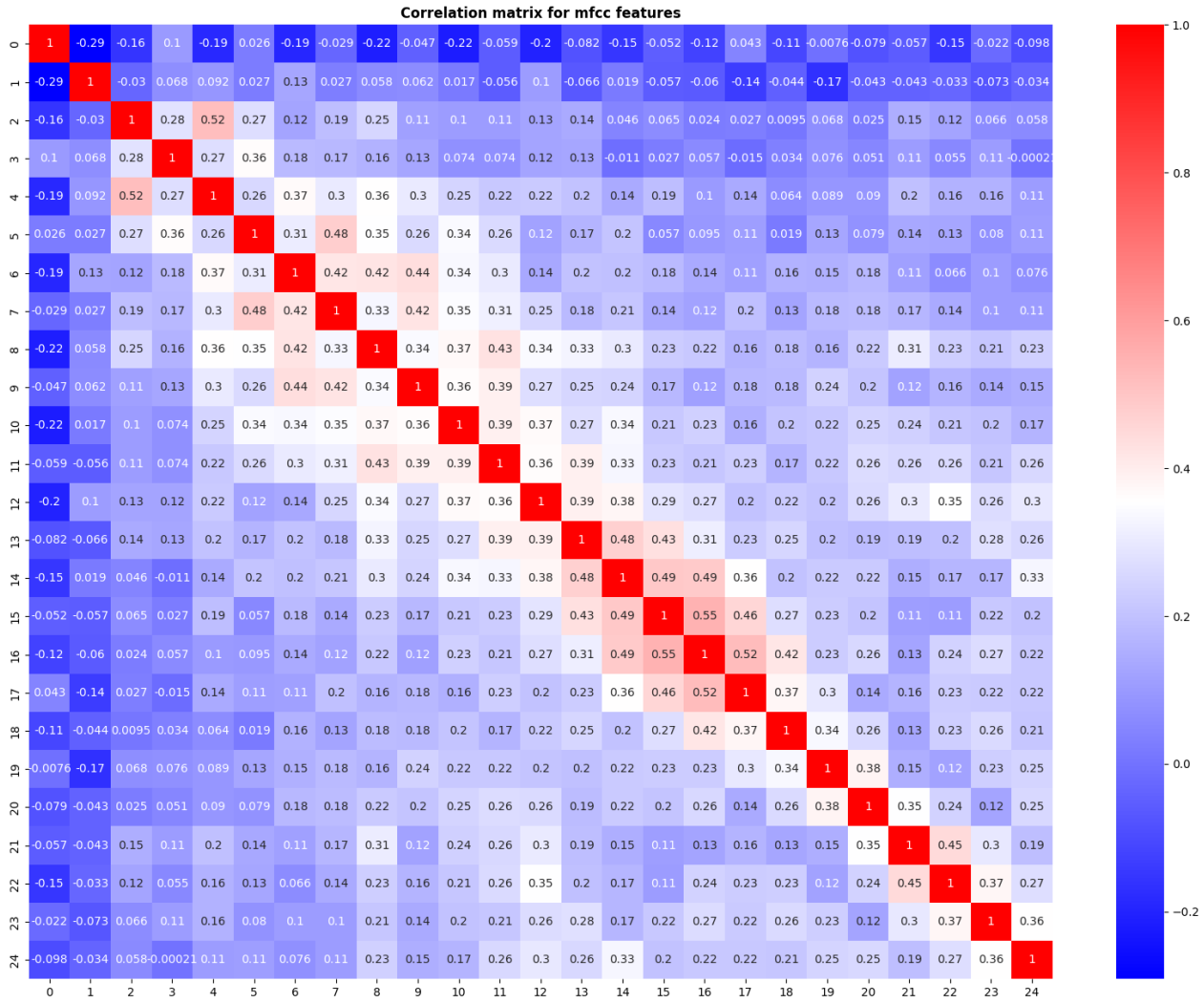


Figure 4: Correlation matrix for mfcc audio features

Our first model uses a **MLP** neural network. The input shape is **25**, corresponding to number of mfcc features we extracted. We use **3 dense layers** with **512**, **256** and **128** neurons respectively. The output shape is **10**, since we classify samples to one of 10 classes. To prevent overfitting, we also add **dropout layers** with drop rate of **0.2**. For activation functions across all hidden layers, **ReLU** is used, the output layer uses **Softmax** since we are dealing with a classification problem. For loss function **Categorical crossentropy** is used and for optimizer **Adam** gave best results. During training early stopping is also used, to stop training when accuracy no longer increases.

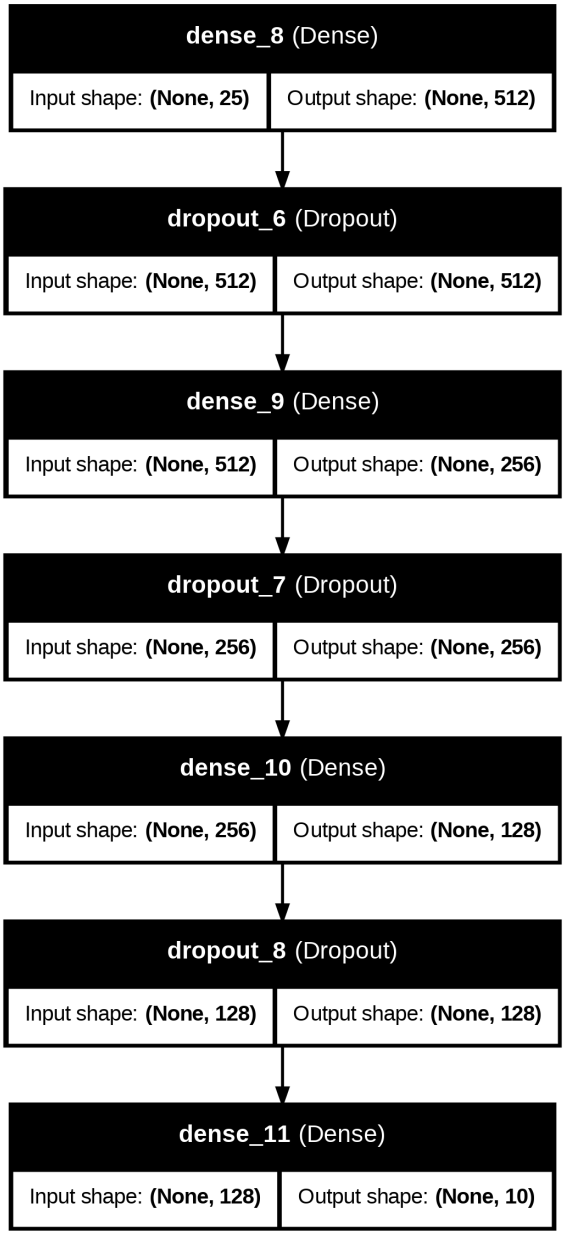


Figure 5: First approach model

Tested over 5 split cross validation, the first approach succesfully managed to classify urban sounds with an average accuracy of **91.54%**

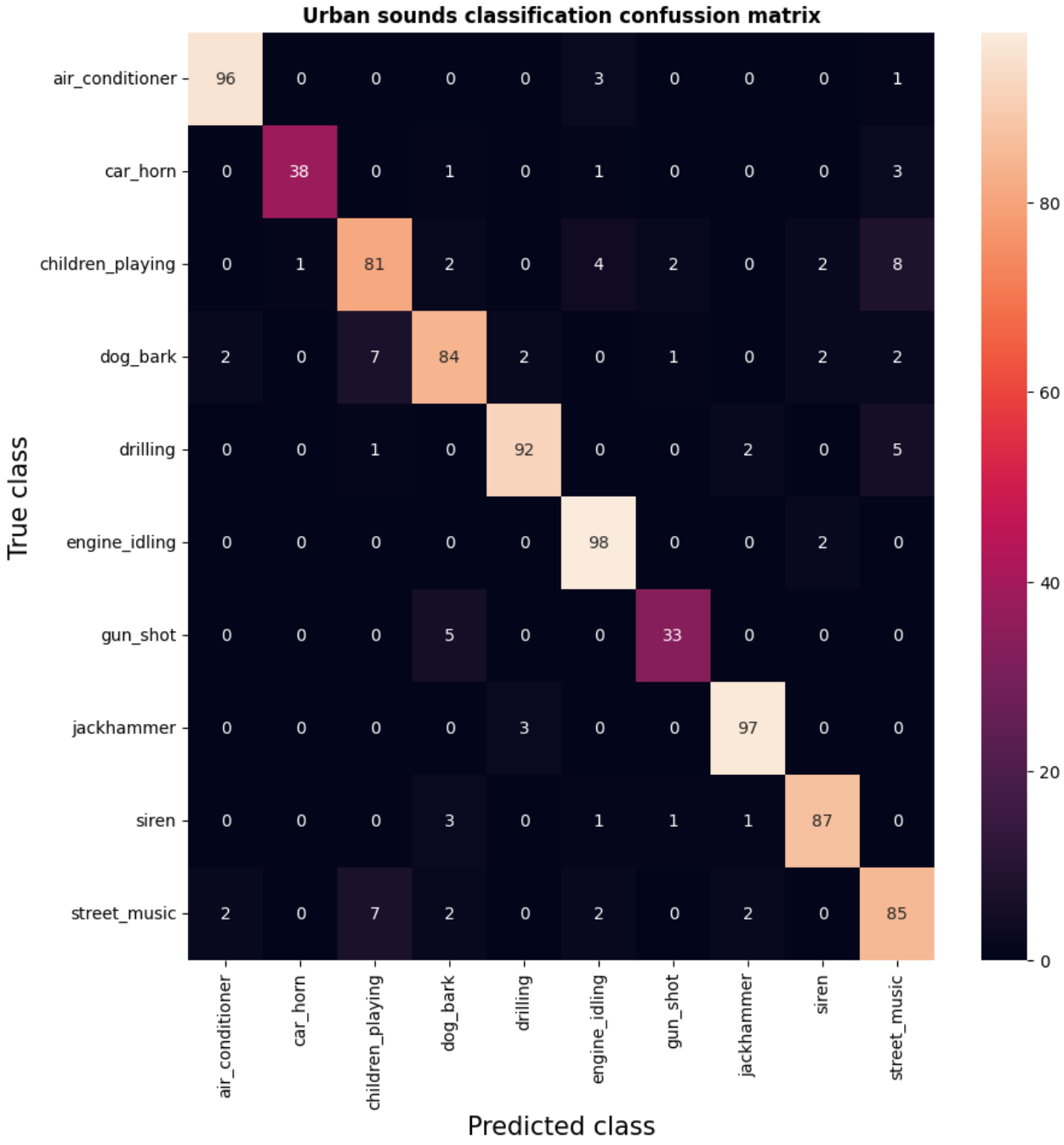


Figure 6: Confusion matrix for first approach

2.4 Approach 2 - Convolutional Neural Network

For the second approach, a convolutional neural network is used. For each audio sample, a mel spectrogram image is created. The images are then used to train the network, as each type of sound has a distinct mel spectrogram, the network can use the images to classify them. In this approach, small **80x60** images are used. This helps to save on memory, since larger images don't improve accuracy in measurable way, so we don't lose any accuracy on our model.

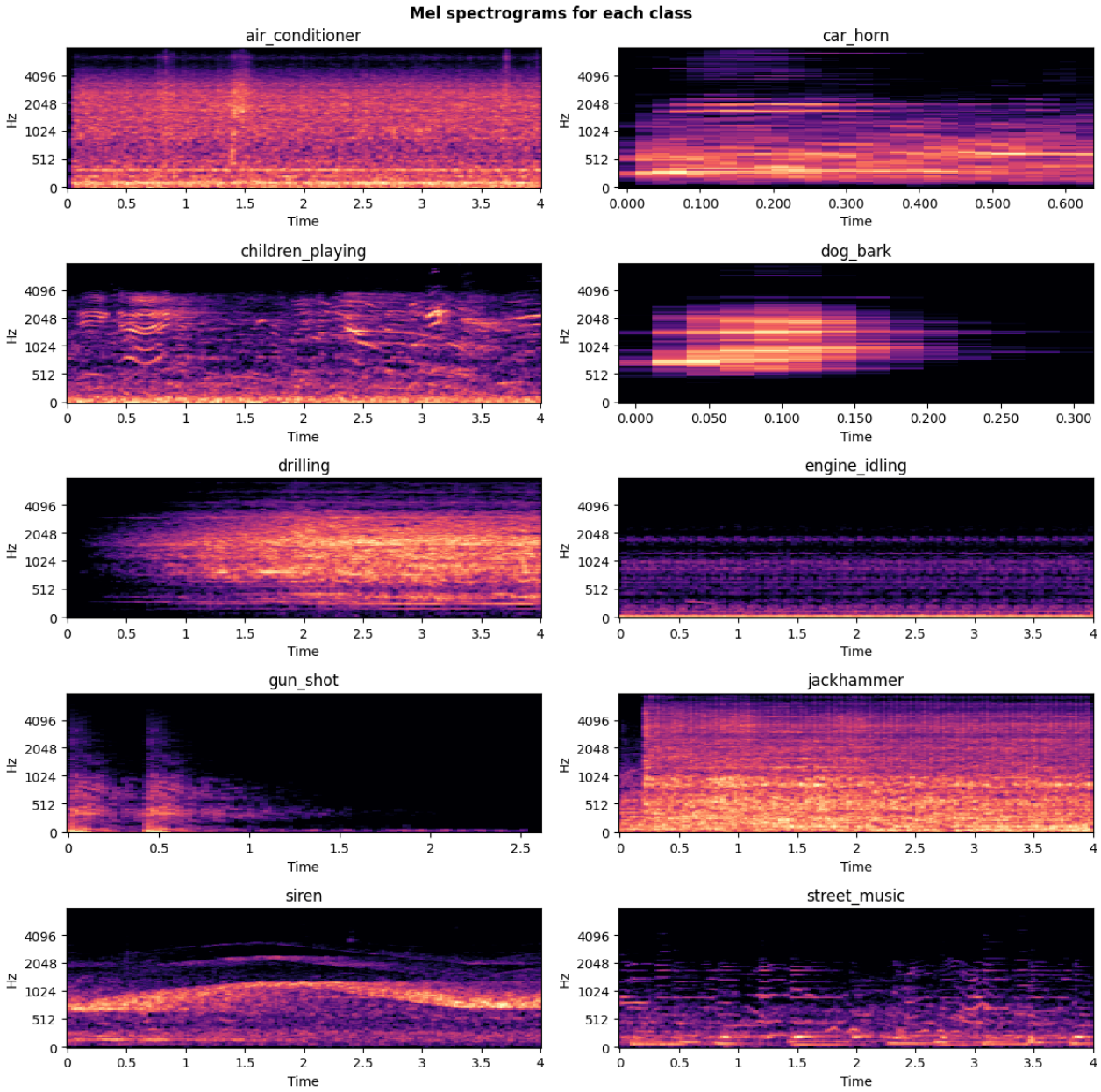


Figure 7: Example of a mel spectogram for each class

Data was split into 80-10-10 split, 80% for training, 10% for valdiation and 10% for testing. The network uses two **2D convolution layers**, with **3x3** kernel size and **16**, **32** filters, a **dense layer** with **512** neurons and **ReLU** activation as well as **3 dropout layers** with 0.3, 0.3 and 0.5 dropout rate respectively. Adam optimizer with learning rate value of **0.00005** is used, which helps to prevent overshooting of optimal solution. While there is a possibility that there are better ways to design the network, testing them is very time consuming and this network performs the best out of the ones tested.

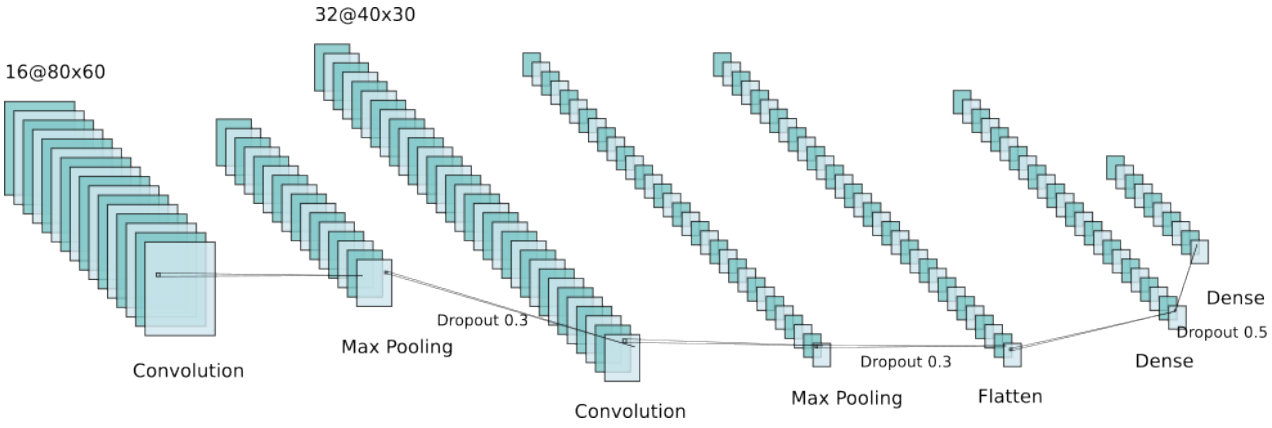


Figure 8: Convolutional neural network model

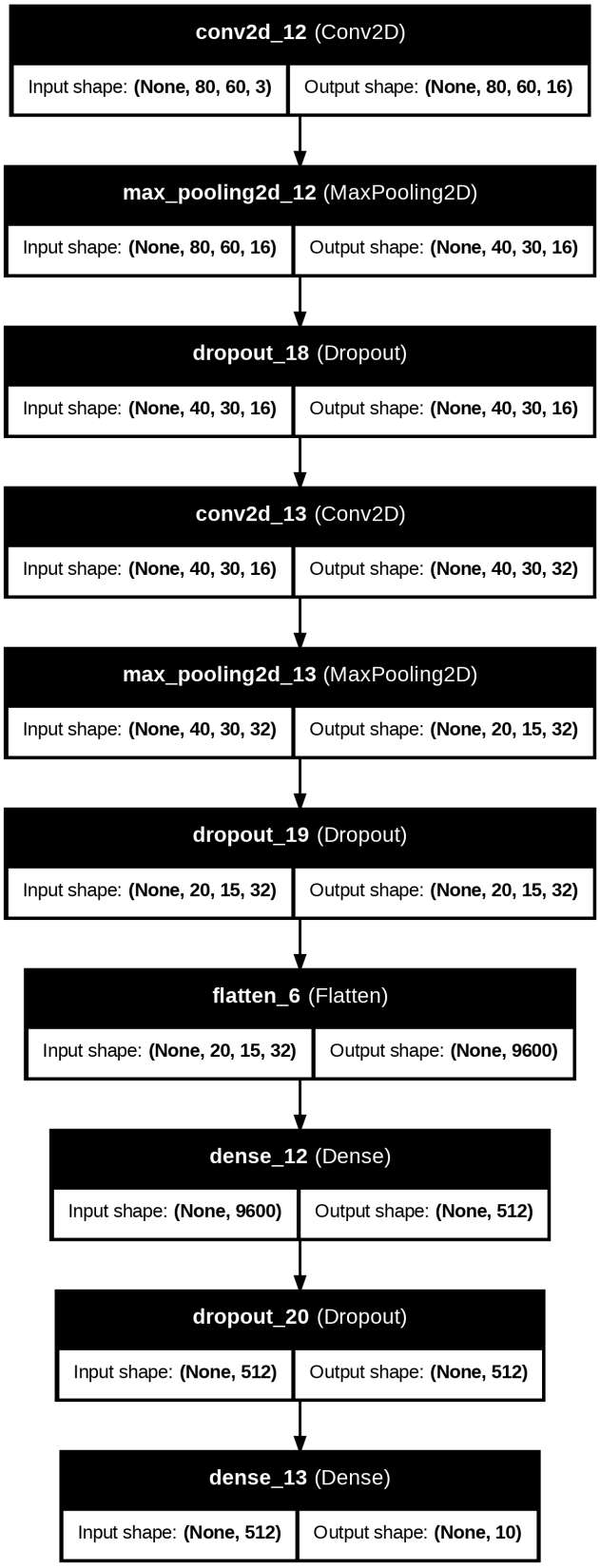


Figure 9: Convolutional neural network model

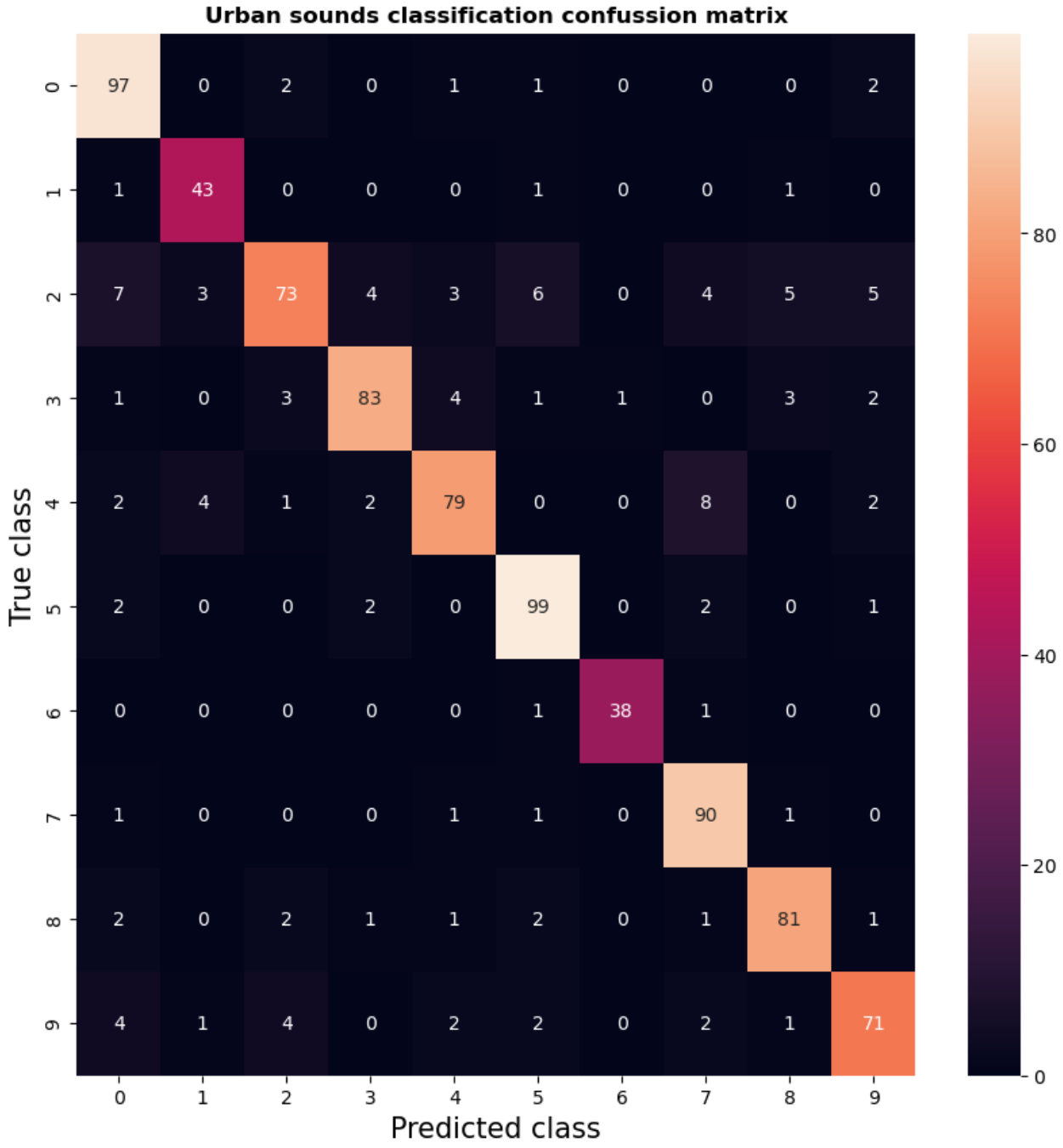


Figure 10: Confusion matrix for the second approach

Tested over 5 split cross validation, the second approach is also successful, managing to get an average accuracy of **86.29%**.

2.5 Performance - Comparision of both approaches

Both approaches were evaluated based on accuracy of predictions, speed of training and speed of predicting new samples in real-time

Traning accuracy and loss

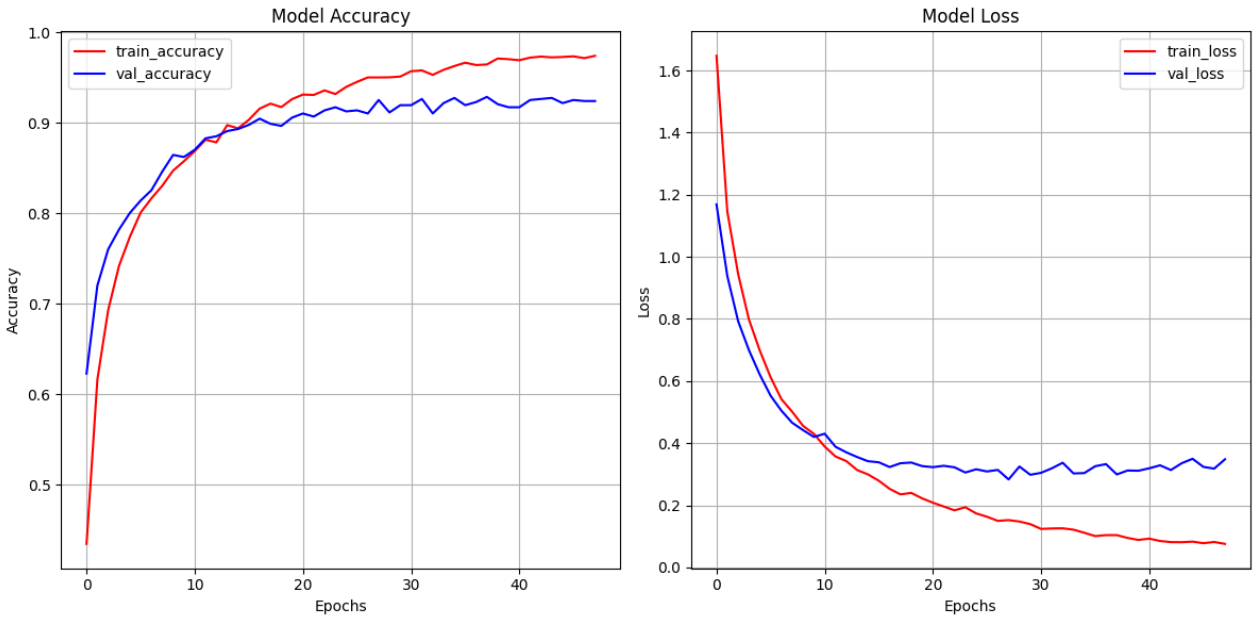


Figure 11: Training of the multilayer perceptron network

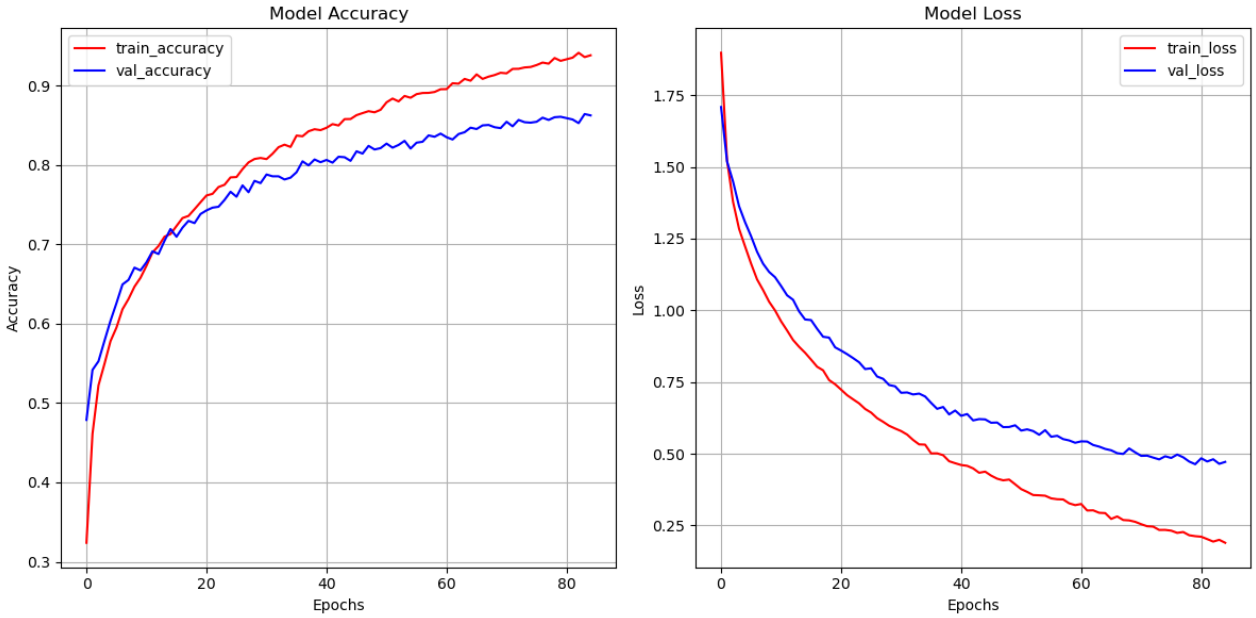


Figure 12: Training of the convolutional neural network

Average accuracy from cross-validation:

- Multilayer perceptron network - **91.54%**
- Convolutional neural network - **86.29%**

Average training time:

- Multilayer perceptron network - **1 minute 2 seconds**
- Convolutional neural network - **13 minutes 25 seconds**

Average real-time prediction time:

- Multilayer perceptron network - **200ms**
- Convolutional neural network - **260ms**

3 Summary

3.1 Directions of development

- Increasing the model's accuracy - further experimentation with the design would likely result in even better performance of the model, however, this requires more powerful hardware in order to train the network.
- Training model to recognize new classes - by using Mel spectrograms for classification, it is possible to train the network to recognize new classes, as each type of sound is associated with a distinct Mel spectrogram.

3.2 Conclusion

- Classification of audio is a complex task that requires large amount of data
- Fine tunning of neural network parameters is crucial to prevent overfitting on training data
- Classification with mel spectograms using convolutional neural networks takes more time and computational power compared to classifing with mel frequency coefficients using multilayer perceptron network