Bachelor thesis

Artificial Intelligence

Radboud University 

Audio classification using GRU

Author:
Luyang Busser
s1032716

First supervisor:
dr. K. van der Heijden
Donders Institute
kiki.vanderheijden@donders.ru.nl

Second supervisor:
dr. H. Fitz
Donders Institute
hartmut.fitz@donder.ru.nl

June 20, 2022

Abstract

How the brain processes real life sound fragments into neural representations is studied actively and there are still many things unexplained. In this paper, inspired by Francl & McDermott (2022) and Van der Heijden & Mehrkanoon (2020), I investigated deep recurrent neural networks (RNNs) with gated recurrent units (GRUs) to come one step closer to understanding the auditory processing in humans. This biological inspired recurrent neural network is trained on predicting the azimuth location of sound as well as predicting the category of sound (i.e. speech, nature, urban, music and human sounds). Both predictions are multi-label multi class classification tasks, and the performance of the model is measured using the binary cross entropy loss. The model is human inspired because of the architectural design choices, such as separate left and right channel input. But also, each classification task has its own pathway, mimicking the different areas in the brain that perform audio localisation and identification. This model was tested using a train/test set of approximately 50,000 one-second audio fragments (approximately 14 hours of audio in total). Additionally, the model was evaluated on an unseen evaluation set to ensure ecological validity. Especially the localisation task of the model showed results that indicate generalisability. It also demonstrated similar error pattern compared to humans, as discussed in the paper. However, the identification task did not show the same results. It did not compare to human accuracy, nor did it have similar error patterns. Overall, the errors measured of this multi-task RNN were bigger than human performance. I suggest in order to conclude more from this human inspired GRU model, one needs to introduce more training data. Another way to extend this research would be by exploring different types of neural networks while staying true to the biological design. For instance, incorporating spiking neural networks (SNNs) into this research and an increase in quantity of the input data is an interesting next step in this field.

Contents

1 Introduction **2**

2 Preliminaries **4**
2.1 Recurrent Neural Networks 4
2.2 GRU 5
2.3 Terminology multi-label audio classification 6

3 Related Work **7**
3.1 AI and optimisation 7
3.2 SELD 8
3.3 Neuroscience and inspiring papers 8

4 Research **10**
4.1 Data set 10
4.2 Network architecture 10
4.3 Training 13
4.4 Evaluation 13

5 Results **16**
5.1 Train/Test set 16
5.2 Evaluation set 18

6 Discussion **21**
6.1 Limitations 22

A Appendix **29**
A.1 Scores predecessor model 1 29
A.2 Scores predecessor model 2 31

Chapter 1

Introduction

Localising and identifying sounds are tasks that we do every second of our lives. These tasks are primary abilities of any living creature and they have been evolving since the beginning of life. Hence, audio localisation and identification are performed extremely quickly and accurately. Within a range of 200 ms people can recognise and respond to sounds [1]. This is necessary, because without the speed or accuracy it will be impossible to survive in the surrounding environment. Localisation of auditory signals is the process of determining the location of the audio source. The brain uses mechanisms like differences in intensity and timing cues to determine this [2]. Sound identification is the process of identifying the type of the audio source, e.g. is it speech?, animal sounds?, or nature? etc. Being able to perform these tasks well means one can (re)act accordingly to sound cues in the environment, think of avoiding dangerous situations, or following a conversation in a noisy area. Because these abilities are so important, sound processing mechanisms of the human auditory system have been researched extensively. I will highlight some of the more relevant researches in this field for this paper below.
In the field of Artificial Intelligence (AI), sound localisation and identification have been modeled for many years and applied in robots and chatbots but also for a better understanding of the human auditory system [3, 4, 5, 6]. Researchers in this field are constantly trying to outperform state-of-the-art models that are trained for audio localisation and audio identification. The best performing models for these tasks are all based on deep learning approaches [7]. This trend can also be seen in recent DCASE challenges [8, 9, 10]. Many of these models are based on convolutional neural networks (CNNs). They require the audio input to be transformed to a spectogram, then processed as an image through the CNN. Despite the high performance of CNNs, these networks variants are not capable of modeling sequences. This leaves room for improvement. However, it is not easy to model acoustic scenes due to their complex sound composites. Research has shown that

recurrent neural networks (RNNs), which are capable of modeling sequences, in combination with CNN are worth investigating, because of the added capability of the RNN layers [11, 12]. The problem with pure RNNs are the high computational complexity as well as the vanishing gradient problem. Hence to my knowledge, there is a lack of prior research into pure RNN performing audio localisation and identification.
I propose a pure RNN with gated recurrent units (GRU) that successfully trains on these two tasks. The GRU solves the vanishing gradient problem, thanks to its memory capabilities, and is therefore a promising start into this topic [13]. The main idea of this paper is to investigate the performance of a human inspired RNN on the audio localisation and identification task. This means that the architectural design choices of the network resemble the human auditory pathway. More specifically, the network has two separate input layers, mimicking the left and right ear. In addition, both the localisation and identification task will have its own branch inside the network, resembling the separate areas that perform the aforementioned tasks in the brain. This topic is inspired by the work of McDermott & Francl (2022) [6] and by Van der Heijden & Mehrkanoon (2020) [14]. Both papers investigate audio localisation using a neural network. However, in contrast to others, they do not try to get the best performing network. They investigate whether a neural network shows human characteristics, when built with similar design choices as the auditory pathway in the brain, or when exposed to similar learning conditions as humans. Some examples of these human characteristics they found are lower localisation accuracy for sounds in the back or side of the head, lower performance when localising sounds concurrently, and sensitivity towards interaural time and level differences.
In short, the field of audio localisation and identification has a sturdy foundation in research, however simultaneous audio localisation and identification by a neural network have not been researched to that same extent. Especially not with a human inspired focus. I add to this field by combining localisation and identification into one human inspired RNN. This research answers the question of whether this human inspired neural network shows similar behaviour as humans when performing audio localisation and identification.
In this paper I first explain some preliminary knowledge about recurrent neural networks in general, then some explanation will follow on GRU and terminology needed for this paper. Followed by a chapter showing related work in this field, I will highlight some important findings and limitations that inspired this paper. Thereafter this paper guides you through a more technical part: the set-up of the research, the dataset, architectural design choices, training procedure and the evaluation step. Finally, I conclude this paper with results, a discussion and notes for future research.

Chapter 2

Preliminaries

The proposed network for the audio localisation and identification is a recurrent neural network with GRU. These type of neural networks are capable of handling sequence data. That is why I propose to work with a RNN that gets raw audio waves as input. A waveform is essentially a sequence of values, which is perfectly suited for a RNN. This chapter will first explain the basics of recurrent neural networks, then the workings and contribution of gated recurrent units in RNNs. Additionally some important terminology of audio classification will be discussed after. After reading this section, the reader will be sufficiently educated to understand the research discussed in this paper.

2.1 Recurrent Neural Networks

Recurrent Neural Networks have been created in the 1980's [15, 16], but in recent years researchers have seen the true potential of these type of neural networks. Because of the temporal dynamics of RNNs, it allows them to predict what is coming in the next time step. That is why RNNs are widely used in natural language processing, predicting financial data, audio classification, weather forecasting and more fields that rely on predictions over time [17, 18, 19]. A RNN is capable of modeling sequence data because it recursively applies a transition function to the internal hidden state vector H_t of the input X. The activation of the hidden unit is calculated by applying a function f on the current input X_t and the previous hidden state H_{t-1}.

$$H_t = \begin{cases} 0 & \text{if } t{=}0 \\ f(H_{t-1}, X_t) & \text{otherwise} \end{cases}$$

You can see that in addition to the current input X_t, a RNN also takes into account the previous time step H_{t-1}. This can be seen as the 'memory'

characteristic of a RNN. Unfortunately, this memory capability also introduces problems. Especially when we are dealing with large sequences, the gradient vector might grow or decay rapidly. This problem is called the 'exploding or vanishing gradient' [20]. That is why it is hard for RNNs to compute long distance relations in a sequence. To solve this exploding or vanishing gradient problem, researchers have introduced a long-short term memory mechanism (LSTM) and gated recurrent units (GRU).

2.2 GRU

Gated recurrent units are essentially the solution to the exploding and vanishing gradient problem. A GRU network is an improved version over the standard RNN [21]. GRU incorporates a so called update gate and a reset gate. These are then two vectors that decide which output to pass on. The reset gate is the one responsible for the short-term memory of the network, also known as h_t. The update gate is the one managing the long-term memory, this gate determines how much of the past information needs to passed on to the future. We can formalise the mathematics of these two gates, initially done by Chung et al. (2014)[21], into the following equations:

$$R_t = \sigma(X_t W_{xr} + H_{t-1} W_{hr} + b_r)$$
$$Z_t = \sigma(X_t W_{xz} + H_{t-1} W_{hz} + b_z),$$

where R_t is the reset gate, Z_t the update gate, and X_t is the input at time t. Remember that H_{t-1} is the hidden state of the previous time step. W_{xr} and W_{xz} are weight parameters, as are W_{hr} and W_{hz}. Finally b_r and b_z are biases. Next, we calculate the following candidate hidden state $\tilde{H}_t$:

$$\tilde{H}_t = \tanh(X_t W_{xh} + (R_t \odot H_{t-1}) W_{hh} + b_h),$$

where W_{xh} and W_{hh} are weight parameters, b_h is bias. $R_t \odot H_{t-1}$ means doing the Hadamard product between the reset gate and the previous hidden state. The tanh function ensures that the output is between -1 and 1. As the last step, the GRU calculates the H_t vector, which holds the information of the current time step and passes it down the network. For that we need to use the update gate Z_t, which determines what to collect from the candidate hidden state and from the previous hidden state H_{t-1}. It is using the following equation for that:

$$H_t = Z_t \odot H_{t-1} + (1 - Z_t) \odot \tilde{H}_t,$$

here we simply determine how much to keep from the old hidden state H_{t-1} and how much the new candidate hidden state $\tilde{H}_t$ is used.
Using these operations a GRU network can store and filter information thanks to the update and reset gates. This eliminates the vanishing gradient problem because the network retains relevant information only and passes it down to the next time steps [13].

2.3 Terminology multi-label audio classification

This research focuses on audio classification in the form of a localisation and identification task. This section will make the reader familiar with some of the more important terminology in order to understand the research steps taken. First of all, this paper in particular looks at so called multi-label classification tasks. Multi-label classification also called multi-output classification is a classification variant where multiple labels may be assigned to each instance. More formally, multi-label classification is finding a model that maps an input $\mathbf{x}$ to a (binary) output vector $\mathbf{y}$. Which means that $\mathbf{y}$ has values of 0 or 1 for each element. And since this is multi-label multiple values of 1 are possible, each input can have multiple labels present. As an example, in this research each audio fragment contains two sounds from different directions. Which means the label vector for audio location has two values which are 1, the rest are 0. The model tries to predict these locations, hence multi-label classification.
Audio location is measured using the term Azimuth angle. Azimuth is an angular measurement often used for localising celestial bodies, but also used in navigation, engineering and ballistics [22, 23, 24]. 0 degree azimuth is defined as in front of the observer. Then it follows a clockwise direction going around the point of the listener. This means that a 90 degree azimuth located sound source is to the right of the observer, 180 degree means behind the observer, 270 degree means to the left of the subject. Often researchers also use -90 to describe the 270 degree azimuth. In this paper, I use the azimuth spectrum from 270 (-90) degree until 90 degree. In short, the front half of the listener's perceptive field. Figure 2.1 shows the localisation spectrum that I use in this research.

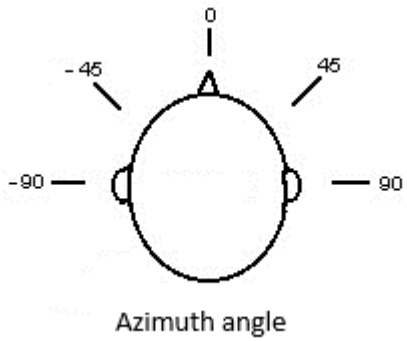

Figure 2.1: Azimuth location

Chapter 3

Related Work

As stated before, deep learning models are currently the state-of-the-art models performance wise, when talking about audio and image processing as well as natural language processing. In this chapter, I focus on deep learning in conjunction with audio processing, since this topic is most relevant to the research question of this paper.

3.1 AI and optimisation

In AI research the goal of these deep learning approaches is mainly to optimise the performance of the model. Think about audio classification tasks, like environmental sound classification, music classification and natural language utterance classification. For these tasks the better the performance the more reliable this system will be when detecting discrepancies in the input data. High performance in the natural language utterance classification task means that a social robot is robust against people talking in various accents and it will still be able to function and deliver the services the user expect. A high performance environmental sound classification model implemented in security systems can detect discrepancies in sound and inform the user to take action, for example predictive maintenance on factory machines [25].
While my research also uses deep learning methods, it does not focus on optimisation of the model to be as high as possible. Instead, it focuses more on finding similarities in behaviour such as error patterns. The model will be tuned to perform close to human performance, so the goal is not to surpass human performance. This study sets up further empirical research to exploring how the human brain performs audio localisation and identification. Even though optimisation is not the main focus of this paper, it is still relevant to discuss.

3.2 SELD

As mentioned, this paper investigates audio localisation and identification. One of the more closely related fields is the SELD, Sound Event Localisation and Detection task [26]. Also, as mentioned before the state-of-the-art models used for this topic are all based on deep learning approaches. Products that apply the SELD are using it for navigation without visuals or occluded targets, self-localisation, for inference of the environment, smart home applications, audio surveillance, among others [27, 28]. Furthermore, a yearly contest is held, known as DCASE, which challenges people to build a model that outperforms other models while performing this SELD task. The top models of this year's DCASE challenge were based on pure RNN or a CRNN. The localisation errors of the best performing models are around 10 degrees [29, 30]. The winner of the challenge had an error as low as 8.5 degrees. It shows that the deep learning approach to audio localisation and detection is highly promising.

3.3 Neuroscience and inspiring papers

In the field of neuroscientific research, deep learning models are also explored extensively. The papers that inspired this research in particular are using neural networks to better understand the auditory pathway in humans. More concrete, Francl & McDermott (2022) looked at how deep neural networks trained on sound localisation perform in real life environments. They want to see how simulated real life, reverberation, noise, multi-source environment, among other alterations, affect the performance of localisation on their network. They compared human participants with their own neural network and see if there are similarities in performance loss, or whether the network reveals similar error patterns as humans. They found, under these circumstances, that the model exhibited many characteristics of human spatial hearing. These characteristics were: sensitivity to monaural spectral cues, interaural time and level difference, integration across frequency, limits on localising of concurrent sources and biases for sound onsets. They illustrate that artificial neural networks can reveal the constraints that shape our perception and thus explain how our auditory perception adapts to real-world environments [6].
Van Der Heijden & Mehrkanoon (2020) developed a biological inspired neural network to uncover some neurocomputational mechanisms in the brain. They built the neural network to resemble the early stages of human auditory pathway and found that it could predict azimuth location of sounds. They tested the model with different loss functions, namely the mean squared error (MSE) and angular distance (AD). They found that although the overall prediction errors were bigger than humans, the error patterns were similar.

They discussed that in future work researchers should try different neural network architectures [14]. Which is something I follow up on in this paper. As shown in this chapter, the field of audio classification and localisation using deep neural networks is not new. Many researchers are making progress within this topic with many different approaches. The two works that inspired me to do my research in particular focus more on biological inspired neural networks and uncovering the complexities behind human auditory system. They conclude that more work is needed with different types of architectures and different audio processing tasks are needed to progress this research further, which is precisely the goal of this research.

Chapter 4

Research

4.1 Data set

The data set used in this research consisted of real-life sound clips with a duration of one second, and are sampled at 16kHz. The two channel audio clips have two sources of sounds spatialised into nineteen locations which covers the frontal azimuth range, i.e. the segment from 0° until 90° and the segment from -90° to 0°. The elevation is constant at 0°. The locations are binned as intervals of ten degrees. Therefore, we have 19 location bins.
Additionally, each stereo sound fragment belongs to two identification classes. In this research we explore five different identification categories: speech, nature, music, human, urban. Each audio file contains two sound sources paired with every other location, this results in 171 possible location combinations. A pair of audio fragments consists of two different sound categories, resulting in 10 possible category combinations. Together, this results in 1710 category-location pairs. Each of these unique pairs contains 30 examples, thus we have $30 \cdot 1,710 = 51,300$ two-source sound scenes. I split each audio fragment by their input channel, since the RNN takes the left and right channels separately. The training set is fed into the network in batches of 32, the same holds for testing.
Then, there is an evaluation set which will not be used until the model is tuned extensively based on the performance on the data set mentioned before. It is generated similarly as the train/test set, except it will contain new sounds the model has not seen. This evaluation data set will be used to verify the generalisability of the model, since it has never seen this data before, it eliminates performance bias due to over-fitting.

4.2 Network architecture

The first network design proposed is a GRU network that resembles the first stages of the human auditory pathway. Separate left and right chan-

nel input layers are implemented, which then connects to their own GRU layer. Dropout layers are applied after most layers. This is done to avoid over-fitting problems. Dropout changes, with a set probability, certain values of units from the layer to 0 [31]. By doing this in each iteration we avoid the network being over-fit to the data set. Next, the two branches are concatenated along the sequence length axis and passed through a shared GRU layer. Again, a dropout is applied on this layer's output. Finally, the network splits into two branches, one branch for the identification task and the other performs the localisation task. This splitting represents the seperate areas in the brain that perform these tasks. Each branch consists of at least one GRU layer and dropout and then, after flattening, feeds into a fully connected layer.
Next, in order to turn the output vectors into probabilities, I apply a sigmoid function. This ensures that every value is between 0 and 1. Since both tasks are multi-label in nature, we can predict multiple classes and therefore sigmoid is the preferred function.
Based on test metrics I modified the base architecture to increase performance, while also still respecting the auditory pathway representation. In the subsequent architectural variants, I experimented with the depth, namely, stacking more GRU layers in the localisation branch of the network. In the end the network that was most promising had three more GRU layers in the localisation branch after splitting. It still had one GRU layer in the identification task. This model is not overly complicated (it does not overfit too much), but it performs better in learning the localisation task (the loss decreases more). The network now has three times more GRU layers in the localisation branch compared to the identification branch. The reason for this change is because the localisation task predicts 19 classes, and it is therefore a more complicated task compared to the identification of the sounds. The network needs more layers in order to learn it well. In figure 4.1 we can see the final architecture of the model.

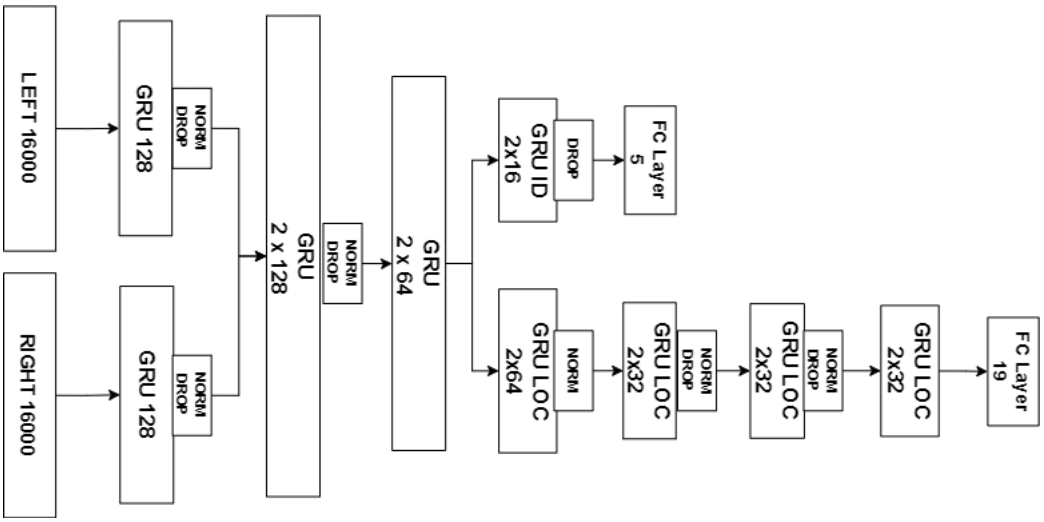

Figure 4.1: Model architecture

Additionally, as mentioned a dropout layer is added after most layers, I experimented with different dropout probabilities. First, the network was tested with similar dropout probabilities, then varying dropout probabilities and finally increasing dropout probabilities. I will discuss the results of these tests in the following sections.
The "width" of the network is another parameter that I experimented with. The amount of hidden neurons at each layer is what is meant with "width". The base value I started with in this research is 512 units. From this value I divide the number of neurons by two after some layers. To reduce the chance of over-fitting, it is usually good practice to keep the model as simple as possible, this phenomenon is called Occam's Razor [32]. Therefore, I do not experiment with values higher than 512. It introduces over-fitting in a larger degree, which is not desirable. I found that even with 512 the model over-fitted too much and hence I found the best models were the ones with 256 units as starting hidden dimension size.
After experimenting with all hyper-parameters mentioned before, I decided to also try layer normalisation. In essence, layer normalisation is similar to batch normalisation, except that the normalisation axis to process the input x is different. Instead of normalising on the mini-batch axis we normalise along the feature axis. The formula for layer normalisation for each layer is as follows:

$$\sigma^l = \sqrt{\frac{1}{H}\sum_{i=1}^{H}\left(a_i^l - \mu^l\right)^2}$$

H is the number of hidden units in the current layer. All the hidden units in a layer share the same σ and μ. This technique of normalisation is commonly

used in recurrent neural networks specifically [33]. I have found that layer normalisation improves training times and generalisability of the model.

4.3 Training

This network is trained using a binary cross entropy (BCE) loss. The network is essentially doing two multi-label classification tasks, and BCE is often used in these classification tasks. Cross entropy calculates the difference in probability distributions. In this case the difference in the true labels and the predicted labels. Because the true labels are one hot encoded, the binary cross entropy variant is used in this research. The formula for BCE looks as follows [34]:

$$H_p(q) = -\frac{1}{N}\sum_{i=1}^{N}\left(y_i \log(p(y_i)) + (1-y_i)\log(1-p(y_i))\right)$$

where y is the true label and $p(y)$ the predicted probability of that specific class for all N data points. The audio fragments are split into a 80%/20% train-test split. The network is trained using the Adam optimiser with weight decay with a learning rate of around 0.002 and early stopping. The implementation of the network is in PyTorch (version 1.11) [35]. Tensorflow and Seaborn are used for parts of the visualisation of the evaluation metrics. Because of the multi-task nature of this network, it outputs two cross entropy losses. However, since the localisation task is more difficult to train, I assign a higher weight to the respective loss when backpropagating. If I assign an equal weight to both, the network will learn the (easier) identification task well while leaving the localisation task behind, performance wise. During training I also tested on various weight ratios to see which weighting works best.
Then, the model also is implemented with early stopping. It is implemented as follows: after training one epoch, the model tests on the test set and calculates the weighted average loss. If the difference between the previous test loss and the current weighted test loss is below a threshold (delta) it will increment the trigger value. This trigger value is also incremented when the current test loss is higher than the previous test loss. If the trigger value reaches the patience value, the model stops training.

4.4 Evaluation

The model is evaluated using a seperate dataset which it has never seen. This ensures that the model is generalisable and not overfitting to the train/test data set. I check the performance of the model based on the evaluation loss, F1-score, confusion matrix and average angular error for the localisation

task, and multi-label confusion matrix for the identification task. These will also be shown in the results section. The F1-score is defined as follows [36]:

$$F1 = 2 * \frac{Precision \cdot Recall}{Precision + Recall}$$

Precision calculates the correct positive predictions of every positive prediction. In formula form:

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

Recall calculates how much of the total number positives did the model predict as positive. In formula form:

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

So the F1 score combines both precision and recall, this score captures the overall performance of our current classification tasks well.
For the evaluation of the localisation task I show a confusion matrix. This is a clear way to visualise the amount of correct predictions and also prediction errors. A perfect confusion matrix has a only values on the diagonal. Which means that the y-axis (the true labels) are predicted correctly (the x-axis). I refer to figure 4.2, this figure shows the perfect confusion matrix. As can be seen, there is a clear diagonal line.

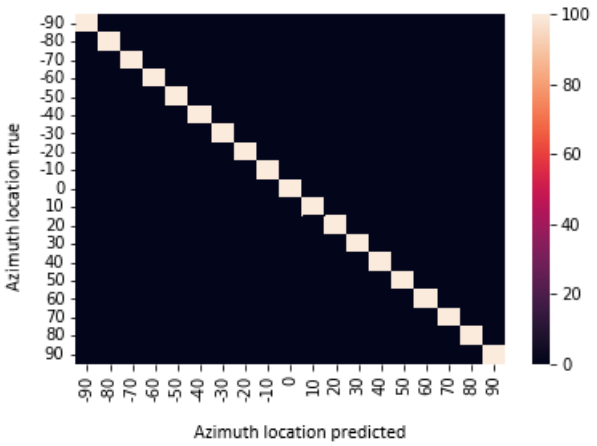

Figure 4.2: Perfect confusion matrix example

The identification task is also a multi-label classification task, except there is no order between the classes as with localisation. Therefore a standard confusion matrix will not tell us much. There is an alternative to evaluating these type of tasks called multi-label confusion matrix (MCM).

The multi-label confusion matrix is a collection of confusion matrices for each class. In this case we have five classes, thus five confusion matrices. In MCM the $MCM_{0:0}$ is true negative (TN), $MCM_{0:1}$ false positive (FP), $MCM_{1:0}$ false negative (FN) and $MCM_{1:1}$ true positive (TP).
Lastly, I look at the average angular error of the localisation task. This is closely related to the corresponding confusion matrix of the respective task. For each location-bin I calculate the average distance of all predictions to the target. Then these numbers are summed and divided by the number of classes to obtain the (approximate) average angular error.
As a final remark I refer to Appendix A to gather more insight in other models I trained using the procedure described above. Earlier models had different parameters and different architectural design choices. Their performance was rather good on the train/test set, however performed significantly worse on the evaluation set. These models were not good enough to be included in the main text, but are still part of the research process. Therefore, I included them in the Appendix section.

Chapter 5

Results

In this chapter, I will show the results from training, testing and evaluation. The results of the best-performing model are shown here. However there are a dozen models preceding the one that is shown. I refer to appendix A for information on the hyper-parameters set-up and performance of some of these parent models.

5.1 Train/Test set

First, the model is trained and tested on the train/test data set, of which the creation is discussed previously. The BCE losses of the training, generated via Tensorboard module of TensorFlow (version 2.9.0) [37], are visualised in figure 5.1. Since the model performs two task, we have two loss graphs. The first one shows the train/test losses of identification (titled Train/test loss id) and the second graph is the localisation training/test loss (titled Train/test loss loc).

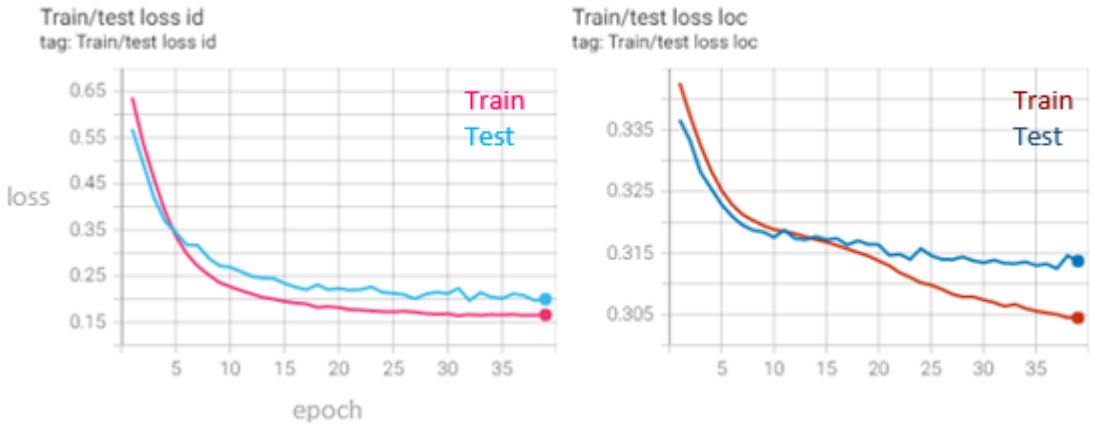

Figure 5.1: Train/test losses

Then, two types of confusion matrices are shown. First a regular confusion matrix of the localisation task during training is illustrated in figure 5.2. The colour bar represents the proportion value of the frequency after l2 normalisation. Then, the multi-label confusion matrix for the identification task, which is generated by scikit-learn's (version 1.1.1)[38] multi-label confusion matrix method, shown in table 5.1.

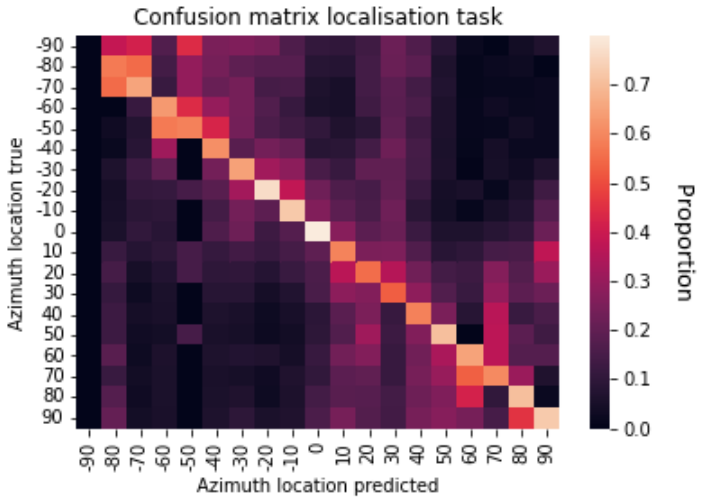

Figure 5.2: Localisation confusion matrix

Remember that per class there is a 2x2 matrix shown. As stated before, in the MCM the $MCM_{0:0}$ is true negative, $MCM_{0:1}$ is false positive, $MCM_{1:0}$ is false negative and $MCM_{1:1}$ is true positive. It is normalised using the scikit-learn's normalization method using the l2 norm.

Human	0.99987329	0.04761252
	0.01591888	0.99886588
Music	0.99967995	0.03214106
	0.02529802	0.99948334
Nature	0.99935684	0.02928138
	0.03585967	0.99957121
Urban	0.99978868	0.03639393
	0.02055734	0.99933752
Speech	0.99978025	0.03423518
	0.02096287	0.9994138

Table 5.1: Multi label CM test set

In addition I also show the summary scores, F1 score, BCE loss, and average angular error which can be seen in table 5.2.

Task	F1 score	BCE loss	avg angular error
Identification	0.9617	0.0858	-
Localisation	0.1819	0.2976	33.6

Table 5.2: Summary scores training phase

Each task has its own F1 score and loss score therefore two rows with values are shown.

5.2 Evaluation set

The model was evaluated on the unseen evaluation set, generated as mentioned previously. First, figure 5.3 shows the localisation matrix.

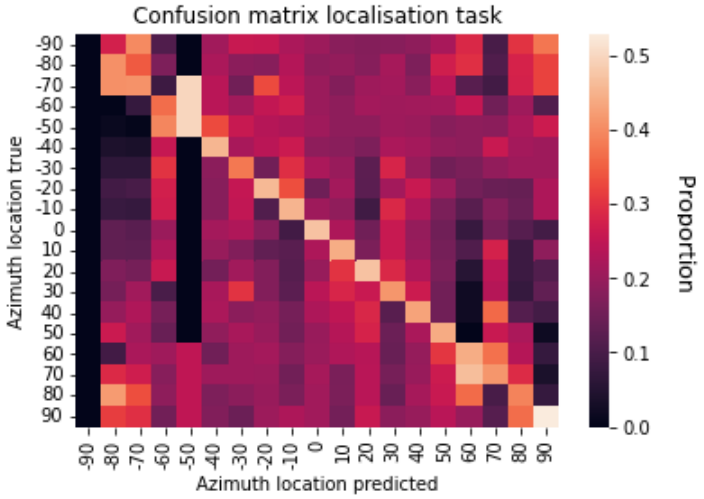

Figure 5.3: Localisation confusion matrix evaluation set

Then, the multi-label confusion matrix is illustrated in table 5.3, with similar labels as mentioned earlier. Remember, there are five 2x2 confusion matrices, each category has one associated matrix.

Human	0.82380961	0.84162048
	0.56686658	0.5400694

Music	0.83575033	0.82659925
	0.54910963	0.56279098

Nature	0.83182303	0.83245517
	0.55504093	0.5540924

Urban	0.8290453	0.83646644
	0.55918144	0.54801816

Speech	0.83321379	0.83010688
	0.55295097	0.55760432

Table 5.3: Multi label CM evaluation set

Table 5.4 shows summary scores during evaluation of the model; the F1 scores of both tasks, BCE loss, as well as the average angular error of the localisation task.

Task	F1 score	BCE loss	avg angular error
Identification	0.3888	1.8251	-
Localisation	0.0730	0.3763	52.5

Table 5.4: Summary scores evaluation phase

Chapter 6

Discussion

The loss graphs in figure 5.1 show that during training, the identification task learns fast, while the localisation task descends slower. Additionally, the train/test loss of the identification follows a similar descend pattern. However, the training loss graph of the localisation descends faster after approximately twenty epochs compared to the test loss. It seems the model starts over-fitting slightly, because the performance of the test set is not improving as much.
From the summary scores we can see that identification performs well, with a loss of 0.0858, and a F1 score of 0.9617. Localisation performs worse compared to the identification, with a loss of 0.2976 and F1 score of 0.1819. These scores are lower also because the localisation task has nineteen classes to predict. While identification only has five. The localisation task is harder to predict because of the increased amount of alternative classes.
If one looks at the scores during evaluation of the model, it can see that the identification is performing worse than localisation. This is another indication that the model was over-fitting on the training set. The localisation task performed worse than during testing, it has a BCE loss of 0.3763, the localisation confusion matrix shows the diagonal line. This means that the model can perform the localisation task on an unseen test set to an extent, it has the potential of being generalisable.
An interesting phenomenon that can be deduced from the localisation confusion matrices is that the model predicts the sounds in front better. The sounds on the sides are predicted with less accuracy. This is congruent with the findings of Oldfield et al.[39] and also from the papers mentioned before such as Francl & McDermott's [6], they also found that accuracy of audio localisation in humans is more accurate in the front and the least accurate on the sides and in the back.
The identification task seems not the favour a particular class or cluster of classes. Each class' prediction have approximately the same true positive/negative and false positive/negative rate. From the evaluation scores

not much can be concluded as it performs close to chance level when predicting on this data set.
Since this is to my knowledge the first biological inspired pure RNN network trained on audio localisation and identification, it is hard to directly compare it with other models. An interesting research to mention here is by Vecchiotti et al. (2019)[40]. They trained a convolutional neural network to perform audio localisation. Similar to this research they also fed raw waveforms into the model. Their WaveLoc-CONV had around a 40 degree angular error, this in the same range of error as the model investigated in this paper. Their other tested models performed much better with angular error of around 8 degrees. They discussed that the next step would be to combine localisation and identification in the same model to see how it performs.
Another interesting paper by Grondin et al. (2019)[12] used a CRNN for audio localisation and detection. They found that their model outperformed the DCASE 2019 base line system. However this model was purely trained on optimisation of these tasks and has no similarities to human auditory pathway, which makes it challenging to compare it to my RNN model.
Then, there is the paper by Van der Heijden & Mehrkanoon (2020), where they trained a biological inspired neural network. It was trained and evaluated using euclidean distance and angular error. They found much better results in angular error, namely values around 20 degrees. Furthermore, they also found a similar error pattern in that the sounds are localised better in the front and worse the more you go the sides and back.

6.1 Limitations

From the results of this paper I can conclude that the model, with the current hyperparameters and input data, over-fits too much on the identification task. This is observable if you look at the difference in performance during testing versus evaluation. The F1 score and loss value over the test set are rather good, considering that for the F1 score, 1 would be the highest possible and for the BCE loss the value 0 means perfect prediction. However, we see a huge decline in performance in both scores when looking at the evaluation set. For the localisation task over-fitting seems not to be a big issue. The summary scores of the test and evaluation set are not as far apart as for the identification task. The scores for the evaluation set are noticeably worse which could mean the model is not generalisable in its current state. I reckon that this behaviour is mostly because the training set is too small. However, the model shows, especially for the localisation task, that it has potential to show human-like characteristics. To confirm that this model indeed has similar error patterns for the identification task, perhaps a different architectural design is needed, or also, a bigger training set. Cur-

rently, the model over-fits drastically on the identification task. Even when simplifying the model, increasing dropout, and using layer normalisation I could not see much improvement to the overfitting problem.
A second point that I did not look into is how GRU units map onto biological neurons. Do the workings of a GRU resemble the mechanisms inside biological neurons? Are there better alternatives to make the model even more similar to the human auditory pathway? Perhaps by changing some of the layers to different types of neurons, we will see an increase in performance or discover new behaviour that makes the model even closer to humans. A promising starting point are the spiking neural networks (SNNs), these type of networks function with spiking mechanisms similar to biological neural networks [41]. This paradigm can give new insights into the dynamics of the human brain.
Finally, another direction for future research is: How can this model be improved further, whilst forgetting about the biological resemblance? In this research I looked at a pure RNN using GRU trained on both localisation and identification. Introducing convolutional layers will make the model benefit from the high performance in feature extraction using spectograms. Additionally, one could introduce bi-directional GRU (biGRU) and investigate its effects on this multi-task set up. Until now biGRU have been widely used for speech and text classification [42, 43]. The audio classification tasks will be an interesting addition to the field while also staying true to the power of biGRUs. I expect it to perform as good if not better because of the additional attention mechanism that will be taken into account while predicting the classes. Investigating these types of research questions will deviate away from the biological inspired constraint, but nevertheless are interesting from an Artificial Intelligence perspective. The possibilities are vast.

Bibliography

[1] Kirill V. Nourski, Mitchell Steinschneider, Bob McMurray, Christopher K. Kovach, Hiroyuki Oya, Hiroto Kawasaki, and Matthew A. Howard. Functional organization of human auditory cortex: Investigation of response latencies through direct recordings. *NeuroImage*, 101:598–609, 2014.

[2] Rickye S Heffner and Henry E Heffner. Evolution of sound localization in mammals. In *The evolutionary biology of hearing*, pages 691–715. Springer, 1992.

[3] Ning Ma, Tobias May, and Guy J Brown. Exploiting deep neural networks and head movements for robust binaural localization of multiple sources in reverberant environments. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(12):2444–2453, 2017.

[4] Mary Ellen Foster, Bart Craenen, Amol Deshmukh, Oliver Lemon, Emanuele Bastianelli, Christian Dondrup, Ioannis Papaioannou, Andrea Vanzo, Jean-Marc Odobez, Olivier Canévet, et al. Mummer: Socially intelligent human-robot interaction in public spaces. *arXiv preprint arXiv:1909.06749*, 2019.

[5] Antoine Deleforge and Radu Horaud. The cocktail party robot: Sound source separation and localisation with an active binaural head. In *2012 7th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 431–438. IEEE, 2012.

[6] Andrew Francl and Josh H McDermott. Deep neural network models of sound localization reveal how perception is adapted to real-world environments. *Nature Human Behaviour*, 6(1):111–133, 2022.

[7] Papers With Code. Audio classification, Jul 2021.

[8] Qiuqiang Kong, Iwona Sobieraj, Wenwu Wang, and Mark Plumbley. Deep neural network baseline for dcase challenge 2016. Technical report, University of Surrey, 2016.

[9] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. Acoustic scene classification: an overview of dcase 2017 challenge entries. In *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*, pages 411–415. IEEE, 2018.

[10] Yohei Kawaguchi, Keisuke Imoto, Yuma Koizumi, Noboru Harada, Daisuke Niizumi, Kota Dohi, Ryo Tanabe, Harsh Purohit, and Takashi Endo. Description and discussion on dcase 2021 challenge task 2: Unsupervised anomalous sound detection for machine condition monitoring under domain shifted conditions. *arXiv preprint arXiv:2106.04492*, 2021.

[11] Sławomir Kapka and Mateusz Lewandowski. Sound source detection, localization and classification using consecutive ensemble of crnn models. *arXiv preprint arXiv:1908.00766*, 2019.

[12] François Grondin, James Glass, Iwona Sobieraj, and Mark D Plumbley. Sound event localization and detection using crnn on pairs of microphones. *arXiv preprint arXiv:1910.10049*, 2019.

[13] Yuhuang Hu, Adrian Huber, Jithendar Anumula, and Shih-Chii Liu. Overcoming the vanishing gradient problem in plain recurrent networks. *arXiv preprint arXiv:1801.06105*, 2018.

[14] Kiki van der Heijden and Siamak Mehrkanoon. Modelling human sound localization with deep neural networks. In *ESANN*, pages 521–526, 2020.

[15] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

[16] Michael I. Jordan. Chapter 25 - serial order: A parallel distributed processing approach. In John W. Donahoe and Vivian Packard Dorsel, editors, *Neural-Network Models of Cognition*, volume 121 of *Advances in Psychology*, pages 471–495. North-Holland, 1997.

[17] Dmitrii Babaev, Maxim Savchenko, Alexander Tuzhilin, and Dmitrii Umerenkov. Et-rnn: Applying deep learning to credit loan applications. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2183–2190, 2019.

[18] Yajie Miao, Mohammad Gowayyed, and Florian Metze. Eesen: End-to-end speech recognition using deep rnn models and wfst-based decoding. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 167–174. IEEE, 2015.

[19] Stefan Balluff, Jörg Bendfeld, and Stefan Krauter. Meteorological data forecast using rnn. In *Deep Learning and Neural Networks: Concepts, Methodologies, Tools, and Applications*, pages 905–920. IGI Global, 2020.

[20] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *International conference on machine learning*, pages 1310–1318. PMLR, 2013.

[21] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[22] AZ Hafez, A Soliman, KA El-Metwally, and IM Ismail. Tilt and azimuth angles in solar energy applications–a review. *Renewable and sustainable energy reviews*, 77:147–168, 2017.

[23] WJ Harlin and David A Cicci. Ballistic missile trajectory prediction using a state transition matrix. *Applied mathematics and computation*, 188(2):1832–1847, 2007.

[24] Shaolin Lü, Ling Xie, and Jiabin Chen. New techniques for initial alignment of strapdown inertial navigation system. *Journal of the franklin institute*, 346(10):1021–1037, 2009.

[25] Matthias Auf der Mauer, Tristan Behrens, Mahdi Derakhshanmanesh, Christopher Hansen, and Stefan Muderack. Applying sound-based analysis at porsche production: Towards predictive maintenance of production machines using deep learning and internet-of-things technology. In *Digitalization Cases*, pages 79–97. Springer, 2019.

[26] Sharath Adavanne, Archontis Politis, Joonas Nikunen, and Tuomas Virtanen. Sound event localization and detection of overlapping sources using convolutional recurrent neural networks. *IEEE Journal of Selected Topics in Signal Processing*, 13(1):34–48, 2019.

[27] Romain Serizel, Nicolas Turpault, Hamid Eghbal-Zadeh, and Ankit Parag Shah. Large-scale weakly labeled semi-supervised sound event detection in domestic environments. *arXiv preprint arXiv:1807.10501*, 2018.

[28] Jie Huang, Tadawute Supaongprapa, Ikutaka Terakura, Fuming Wang, Noboru Ohnishi, and Noboru Sugie. A model-based sound localization system and its application to robot navigation. *Robotics and autonomous systems*, 27(4):199–209, 1999.

[29] Thi Ngoc Tho Nguyen, Karn Watcharasupat, Ngoc Khanh Nguyen, Douglas L. Jones, and Woon Seng Gan. Dcase 2021 task 3: Spectrotemporally-aligned features for polyphonic sound event localization and detection. 2021.

[30] Kazuki Shimada, Naoya Takahashi, Yuichiro Koyama, Shusuke Takahashi, Emiru Tsunoo, Masafumi Takahashi, and Yuki Mitsufuji. Ensemble of accdoa- and einv2-based systems with d3nets and impulse response simulation for sound event localization and detection. Technical report, DCASE2021 Challenge, November 2021.

[31] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[32] Pedro Domingos. The role of occam's razor in knowledge discovery. *Data mining and knowledge discovery*, 3(4):409–425, 1999.

[33] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[34] Daniel Ramos, Javier Franco-Pedroso, Alicia Lozano-Diez, and Joaquin Gonzalez-Rodriguez. Deconstructing cross-entropy for probabilistic binary classifiers. *Entropy*, 20(3):208, 2018.

[35] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[36] Cyril Goutte and Eric Gaussier. A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. In *European conference on information retrieval*, pages 345–359. Springer, 2005.

[37] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol

Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[38] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[39] Simon R Oldfield and Simon PA Parker. Acuity of sound localisation: a topography of auditory space. ii. pinna cues absent. *Perception*, 13(5):601–617, 1984.

[40] Paolo Vecchiotti, Ning Ma, Stefano Squartini, and Guy J. Brown. End-to-end binaural sound localisation from the raw waveform. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 451–455, 2019.

[41] Samanwoy Ghosh-Dastidar and Hojjat Adeli. Spiking neural networks. *International journal of neural systems*, 19(04):295–308, 2009.

[42] Changjiang Jiang, Junliang Liu, Rong Mao, and Sifan Sun. Speech emotion recognition based on dcnn bigru self-attention model. In *2020 International Conference on Information Science, Parallel and Distributed Systems (ISPDS)*, pages 46–51. IEEE, 2020.

[43] Srividya Tirunellai Rajamani, Kumar T Rajamani, Adria Mallol-Ragolta, Shuo Liu, and Björn Schuller. A novel attention-based gated recurrent unit and its efficacy in speech emotion recognition. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6294–6298. IEEE, 2021.

Appendix A

Appendix

A.1 Results of model without layer normalisation and 256 starting dimension

This model has similar layer architecture as the final model. Except that this model is wider, it started with 256 hidden nodes, it also has higher dropout probability (step wise increase when deeper into the network, i.e. 0.20 - 0.25 - 0.30 - 0.35 - 0.40). Additionally, this model did not have layer normalisation. Which is one of the causes of this clear over fitting pattern.

Task	F1 score	avg BCE loss	avg angular error
Identification	0.9792	0.0505	-
Localisation	0.2589	0.3179	31.9

Table A.1: Summary scores test set old

Task	F1 score	avg BCE loss	avg angular error
Identification	0.3900	2.1782	-
Localisation	0.0752	0.4264	60.1

Table A.2: Summary scores evaluation set old

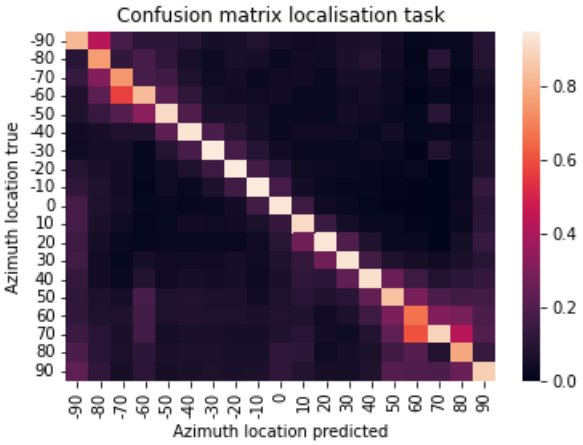

Figure A.1: Confusion matrix predecessor model test set

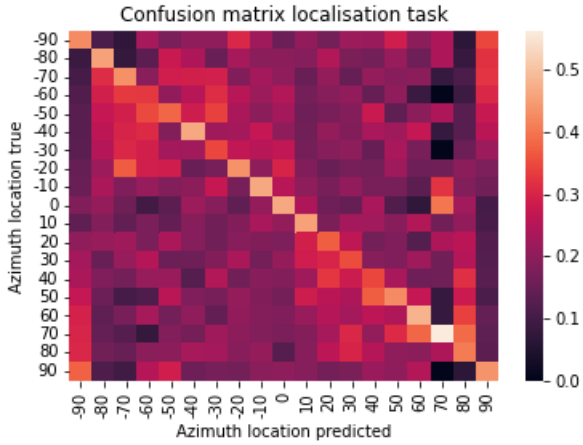

Figure A.2: Confusion matrix predecessor model evaluation set

One can clearly see, due to the much more random confusion matrix, that the model is indeed over fitting. The test set shows a clear diagonal, which indicates high prediction accuracy. However, when looking at the evaluation matrix the predictions are spread over a much wider array.

```
array([[[0.99990889, 0.01861182],
        [0.01349893, 0.99982679]],

       [[0.99991529, 0.02186356],
        [0.01301587, 0.99976096]],

       [[0.99981897, 0.0138395 ],
        [0.01902684, 0.99990423]],

       [[0.9999412 , 0.02545356],
        [0.01084392, 0.99967601]],

       [[0.99994325, 0.02145215],
        [0.01065339, 0.99976988]]])
```

Figure A.3: Identification MCM predecessor model test set

```
array([[[0.83322987, 0.83028565],
        [0.55292674, 0.55733808]],

       [[0.82481842, 0.84045735],
        [0.56539771, 0.54187771]],

       [[0.83011611, 0.83556498],
        [0.55759058, 0.54939163]],

       [[0.83401904, 0.82920001],
        [0.55173567, 0.558952  ]],

       [[0.8361071 , 0.82528096],
        [0.54856624, 0.56472235]]])
```

Figure A.4: Identification MCM predecessor model evaluation set

Similar results can be seen in identification MCM. In figure A.3 the scores are rather good, the true positive and negative rate are near perfect. However, when looking at figure A.4 we can see a drastic shift, where the scores almost indicate pure chance. Another indication of over-fitting of the model.

A.2 Model with 128 starting hidden dimension and higher dropout

This model was the first model where I narrowed the model to counter the over-fitting problem. Also, I increased the dropout probabilities to see if that helps. This model still does not use layer normalisation. Unfortunately, as can be concluded from the scores, it needed much more tuning before we can conclude whether this change was a step in the right direction or not.

Task	F1 score	avg BCE loss	avg angular error
Identification	0.9843	0.0409	-
Localisation	0.4103	0.2265	23.5

Table A.3: Summary scores test set old model 2

Task	F1 score	avg BCE loss	avg angular error
Identification	0.3938	2.5889	-
Localisation	0.0817	0.5140	55.6

Table A.4: Summary scores evaluation set old model 2

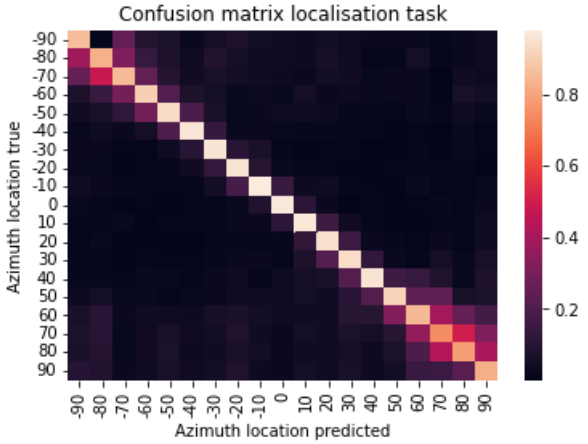

Figure A.5: Confusion matrix predecessor model 2 test set

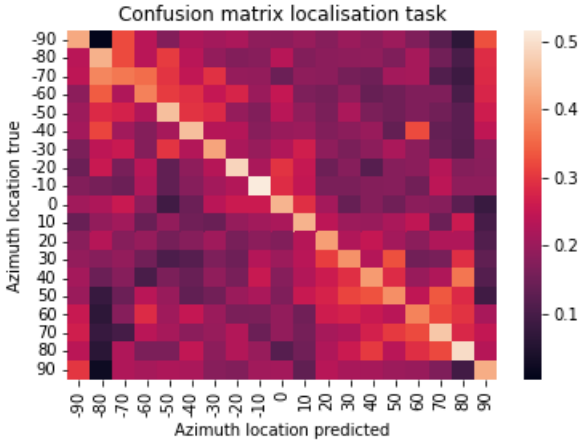

Figure A.6: Confusion matrix predecessor model 2 evaluation set

```
array([[[0.99992506, 0.01576381],
        [0.01224198, 0.99987574]],

       [[0.99995865, 0.02112032],
        [0.00909354, 0.99977694]],

       [[0.99993333, 0.012291  ],
        [0.01154694, 0.99992446]],

       [[0.99996248, 0.01545336],
        [0.00866201, 0.99988059]],

       [[0.99995253, 0.01270049],
        [0.00974355, 0.99991935]]])
```

Figure A.7: Identification MCM predecessor model 2 test set

```
array([[[0.83295093, 0.83055958],
        [0.55334686, 0.55692978]],

       [[0.8304817 , 0.83382092],
        [0.55704592, 0.55203502]],

       [[0.83713545, 0.82370182],
        [0.54699565, 0.5670232 ]],

       [[0.82989913, 0.83514122],
        [0.55791346, 0.55003558]],

       [[0.84172666, 0.81418997],
        [0.53990391, 0.58059857]]])
```

Figure A.8: Identification MCM predecessor model 2 evaluation set

In summary, we can see this clear pattern of good results in test phase, but significantly worse results during evaluation. This is an indication that this model is not generalisable and thus over-fitting on the train/test set.