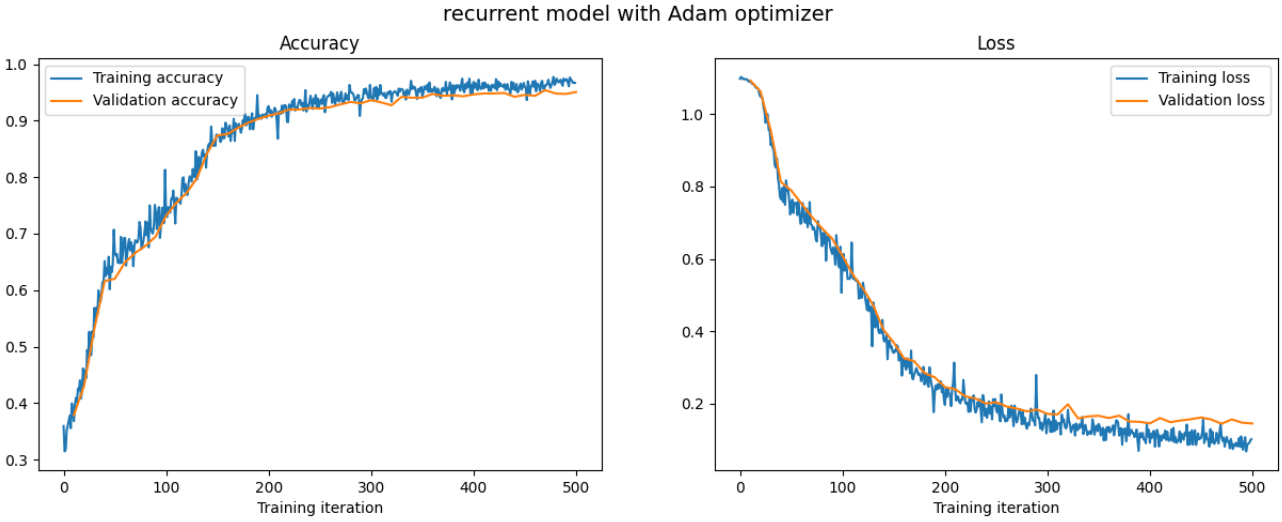
Group number: 530

- Rahabu Mwang'amba rahabum@es.aau.dk
- Yurii Iotov yio@create.aau.dk
- Galadrielle Humblot-Renaux gegeh@create.aau.dk

PhD course Deep learning - assignment 3

Questions

1. **Training curves:** see below. Although the convergence is slower than when training the convolutional & fully connected models, we note significantly less overfitting.



2. **Test accuracy**: We report the recurrent model performance (after 50 epochs with Adam optimization) in the table below. We also include the performance of the previous assignments' models for comparison.

	Fully connected (567171 params)	Convolutional (346067 params)	**Recurrent (70531 params)**
Test accuracy (%)	87.96%	91.78%	**95.69%**

3. **Calculating the number of parameters**:
   GRU units - Let's call I the dimensionality of the input (at a given time-step) and H, the dimension of the hidden state.
   A GRU consists of 3 gates (reset, update and new). Each gate is parametrized by 2 fully connected layers/linear projections (with weight matrix plus a bias term): one which is applied to the input x_t (input size I) and the other which is applied to the previous hidden state $h_{(t-1)}$ (input size H). Both have an output size of H.
   So, for one gate, we have $H(I + 1) + H(H + 1) = HI + 2H + H^2$ parameters
   Since there are 3 gates, for one GRU we have $3(HI + 2H + H^2)$ parameters.

   In our case, the 1st GRU has $I = 40$ and $H = 64$ and the 2nd GRU has $I = H = 64$.
   Filling in, this gives us a total of 20352 parameters for the 1st GRU, and 24960 for the 2nd, totalling to 45312 parameters.

   Fully connected layers - For the fully connected layers, we repeat the same procedure as the previous assignment. The input size of the first fully connected layer is H.

3) for the 1st fully connected layer, we have 128*(64+1) = 8320
4) for the 2nd fully connected layer, we have…128*(128+1) = 16152
5) for the output layer, we have…3*(128+1) = 387

In total, this adds up to 70531 parameters in the model.

4. **Comparison to previous models:** See the table in question 2).
   The recurrent model is the smallest of the 3, with over 5 times less parameters than the convolutional model - the main reason for the low parameter count is the small size of the feature vector fed to the 1st fully connected layer (64 for the recurrent model, vs. 2464 for the convolutional one).
   Despite its small size, the recurrent model achieves the highest classification performance out of the 3 models. Like the convolutional model, it benefits from parameter sharing across the input sequence. However, it is more suited for this task, as it captures the inherently sequential nature of spectrograms, while the convolutional model only captures local structure.