Lab 6 Exercise – Transforming Sequences

Justin Ugwudike (jknu1g19@soton.ac.uk)

April 1, 2023

**1.1 Question 1.1**

When finetuning the ResNet50 model I chose to freeze the gradients on all the previous layers except for the final fully connected layer. The last layer was changed to output only 16 features (rather than 1,000), matching the number of classes present in the smaller dataset I would use to finetune the model.

Since the last layer was changed, it’s gradients would need to be re-learned to classify the new dataset, hence why the gradients on all the other layers are frozen.

The next modification I made to the model was changing the original pooling layer to an AdaptiveMaxPool2d layer since the original pooling layer reduces the pooling window dimensions from 7x7 to 1x1. The AdaptiveAvgPool2d function also reduces the dimensions to 1x1 but for any input window size.

The learning rate used was reduced to a small value (0.00001) to prevent the pretrained ResNet50 model from overfitting to the new data.

To finetune using the feature vectors extracted from the ResNet50 model, the AdaptiveAvgPool2d function is used to average all the feature maps of the ResNet50 model to use as input features for a Support Vector Machine (SVM). Fortunately, I could download already available averaged feature vectors using Numpy to save time (see Appendix A).

The SVM classier was then used to fit the training features to the training labels.

* 1. **Question 1.2**

Finetuning the ResNet50 took three minutes on my PC after which, I obtained a loss of 1.028 (4.s.f) and an accuracy of 67.59% (4.s.f) on the validation data. This is pretty good when compared to the 70.63% (4.s.f) accuracy and 0.7095 (4.s.f) loss I obtained when training the small BetterCNN model in the lab, considering this training also took around 3 minutes and the ResNet50 has nearly 40 more layers.

The SVM on the other hand only took seven seconds to train (much faster that the BetterCNN model and modified ResNet50 model). Additionally, the SVM classifier obtained an accuracy of around 87% based on the printed classification report.

Graphical user interface, text, application

Description automatically generated

Figure 1 – Snippet of an SVM classifier classification report using the ResNet50 feature maps as inputs.

Proving that an SVM classifier with the extracted ResNet50 average layer features is the most efficient and accurate model, not only does it work the best, but it is also the fastest.

1. **Appendix**

Appendix A – the code used to download the averaged ResNet50 features (taken from the feature maps on each layer).

Text

Description automatically generated