Ultrasound Image Classification

Rohini Das

1. **Motivation**

I have always been fascinated by the potential of artificial intelligence and data science to transform healthcare. It is primarily the reason why I have been involved in computational biology and genomics research, affiliated with my university where we proposed a novel miRNA to detect osteoblastic metastasis in prostate cancer (https://onlinelibrary.wiley.com/doi/10.1002/jmr.3042). Additionally, recently I completed a study where we investigated tuberculosis pathogenesis using a comorbidity network (R programming) in order to identify potential therapeutic drugs. Since long I have wanted to apply my deep learning skills to healthcare problems, as it is the cause where I feel I can dedicate my efforts to.

2. **Abstract** (Discuss in brief the method and results for each task attempted)

Task 1:

Following basic exploratory data analysis of image features in the given dataset, three different approaches were undertaken to implement a deep learning-based model capable of classifying anatomical structure in fetal ultrasound images. Pytorch framework was used to first apply pretrained models like ResNet50 (fine-tuned for task relevance), followed by AlexNet and finally a custom CNN model that achieved a competitive accuracy of 86%. The findings emphasized the significance of designing task-specific models, with the custom CNN-approach showcasing its potential to elevate model performance through targeted feature extraction and representation.

3. **Introduction** (Brief discussion about the method opted, details on why and how the model is selected)

Task 1:

- Dataset has 1646 grayscale images which are mostly of the same size.
- Since the dataset is comparatively small, a custom convolutional model would be able to extract the features best.
- Despite this observation, performance on pretrained models was observed to save time.

4. **Data PreProcessing / Analysis** (All pre-processing tasks Resizing, augmentation, etc. and analysis should be discussed)

Task 1:

- The target labels were encoded.
- 224x224 was the chosen input image size, hence resizing transforms were used to scale the images uniformly.
- For pretrained networks that accepted 3 channel images, the images were converted to 3 channels and then trained.
- Images from the directory were loaded as tensors, transformed and then packed as a torch dataset, and finally loaded as batches before passing onto training.

5. **Model Architecture** (Why is the particular model selected? What is the benefit of the selected model in the current task?)

Task 1:

- ResNet50 is aptly suited for efficient transfer learning without the vanishing gradient problem and it has been known to be more efficient for classifying images related to healthcare problems.
- AlexNet was adopted because of the consecutive convolution layers in its architecture that are better for extracting features easily for image recognition.
- Finally, I applied a custom CNN Model composed of 2 convolutional layers and max pooling layers, that finally ended with a softmax probability function for aptly predicting the 4 classes.

6. **Experimental Setting** (Selection method of experimental set up such as but not limited to the optimizer, loss function)

Task 1:

- Cross Entropy loss was applied since it is a multiclass classification problem
- Adam optimizer was chosen since it works best with CNNs and with a learning rate of 0.001 so as to facilitate convergence of the gradients.

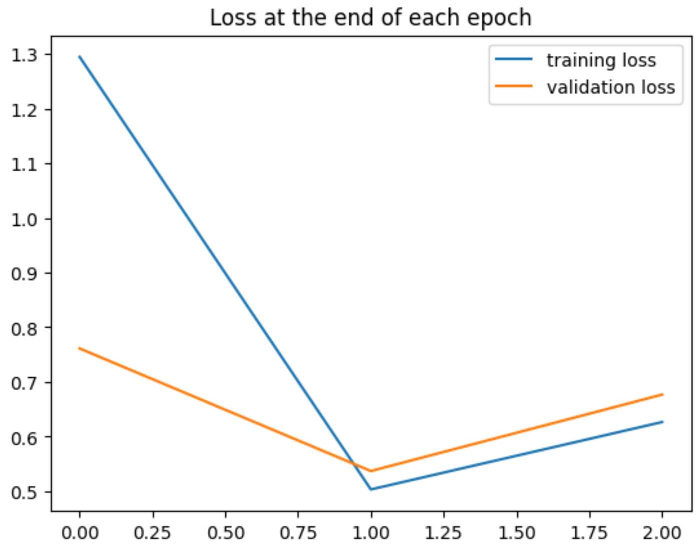
7. **Hypothesis tried** (What other models/modifications you tried to improve the performance of the tasks should be discussed here. The hypothesis can be based on major changes in the pipeline, base architecture, pre-training or so.)

Task 1:

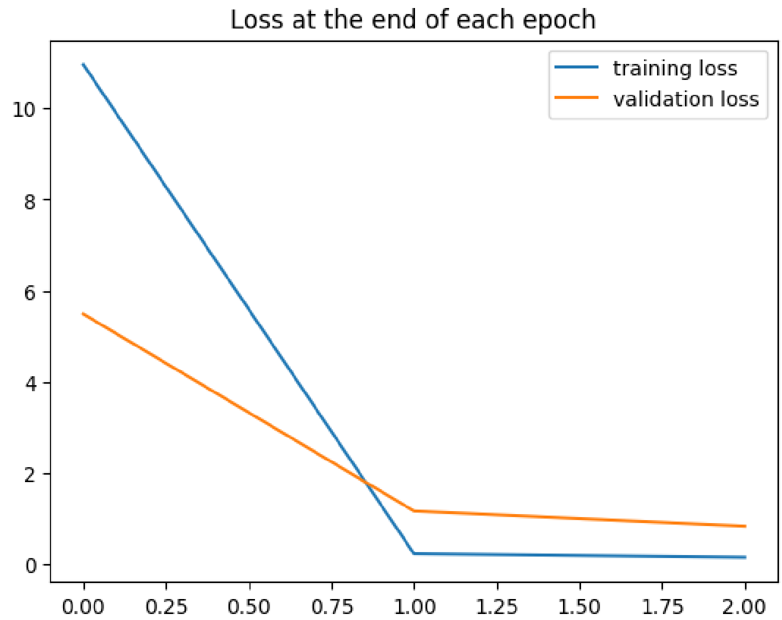
- As discussed in point 5, AlexNet and the custom CNN model were applied to explore alternatives for the ResNet50, since the training time was much higher for ResNet than the other models.
- AlexNet performed comparatively better however because of the 5 consecutive convolutional layers however it is more suited for object detection than 1-channel medical images.

- Finally, the custom CNN model was applied consisting of 2 convolutional layers and max pooling layers so that too much loss of important information from our images did not occur, and that the 4 classes could be predicted seamlessly.

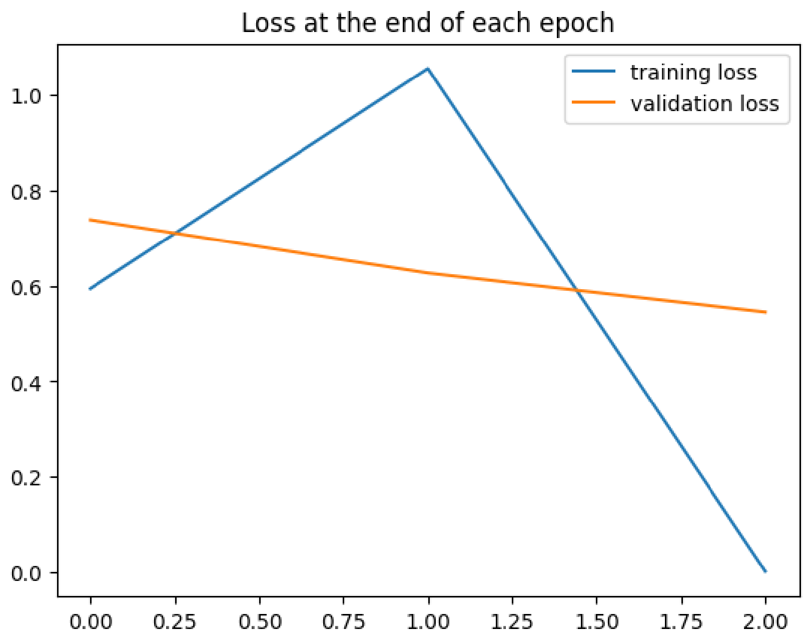
8. **Results** (Critically analyze the results obtained)



For the ResNet50 model, the accuracy hovered in the 74% to 76% range and as depicted in the plotted losses, the model converged somewhere after the 2nd epoch (shown as 1).



For the AlexNet model, the accuracy hovered between 79% to 85%. For the first two epochs, the training loss decreased smoothly however, the model began to slightly underfit after the 2nd epoch as the validation loss increased slightly.



For my CNN model, the accuracy was in the 85% range. By the middle of the second epoch the training and validation losses had converged.

9. **Key findings** (Key takeaways you will have from the results of the best method you opted)

- Pretrained networks can be utilized for a problem if they have been known to do well in that domain (e.g. for transfer learning, medical image classification, etc.) and if our dataset is large enough for training.
- If the model is underfitting but showing promising behavior, then more training data is likely to solve the problem.

10. **Future work** (What are the methods you like to implement that you think will improve the performance.)

- I would like to explore more data augmentation or transforms like RandomFlip or RandomRotate so that more variations of data could be utilized for training.
- I would also like to explore more convolution filters that could possibly highlight the important parts of the input ultrasound images.