Title Detection Report:

The accuracy achieved against the test set:

As mentioned in README.md file, two models are trained to test the performance on the test set. The base Bert model has the AUROC of 0.9707, whereas the Bert that is pre-trained on the classification task has the AUROC of 0.9717. Thus, there are no significant differences in terms of the performance of the models. Also, in terms of training speed, both models converged very fast (within two epochs). More details of the performance could be found in result.txt. The confusion matrix of the two models is also provided.

A brief explanation of your solution including reasons for your model of choice:

The contextual information could be important for this task. Thus, a Bert tokeniser is used to tokenise the input sequences before being loaded into the data. Then, as the Bert has been proved to be so powerful in the NLP domain, a pre-train Bert model is used and fine-tuned for the task. The output of the BERT model is concatenated with the other features such as “IsBold” and “Left”. The concatenated features are then fed into the classifier (a single linear feedforward layer). In my opinion, the classifier now has the advantages of having the powerfulness of the BERT model and having other additional information to make the best decision. A pre-trained Bert model on the sequence classification task was also trained to see if it would be better than the Bert without this extra training. In the end, no significant differences were found, although the latter is less than 0.1% than the former (which occasionally happened, in my opinion).

A description of any improvements you would have made given extra time:

Many further steps could be made to improve the performance if given more time. One could be testing the large Bert (i.e. bert-large-cased) instead of the base Bert (i.e. bert-base-cased). This could result in a better performance as the pre-trained Bert is larger and, consequently, more power. Another improvement could be made by optimising the hyperparameters. For example, we used Adam as our optimiser for the training. Some research shows that AdamW has better performance. We could test them and see if the new optimiser could improve our performance. Also, I had an idea of thinking of the title detection as a visual task instead of an NLP task we had in this case (as we were provided text data). I think that approach would be more reasonable, as humans can detect the titles regardless of the content of the text. For example, I could catch the titles in English pdfs even if I do not speak English. I think the context might not be as important as the spatial information and other information such as font size and colours.

Please also specify the amount of time you have spent designing your solution:

Two nights together for implementation and training. Few hours for writing up the results and cleaning the repository. The first night was wasted because I set the autosave when the best loss was found, which led to overfitting. Thus, the second night was used to re-training and set the autosave when the best AUROC is found.