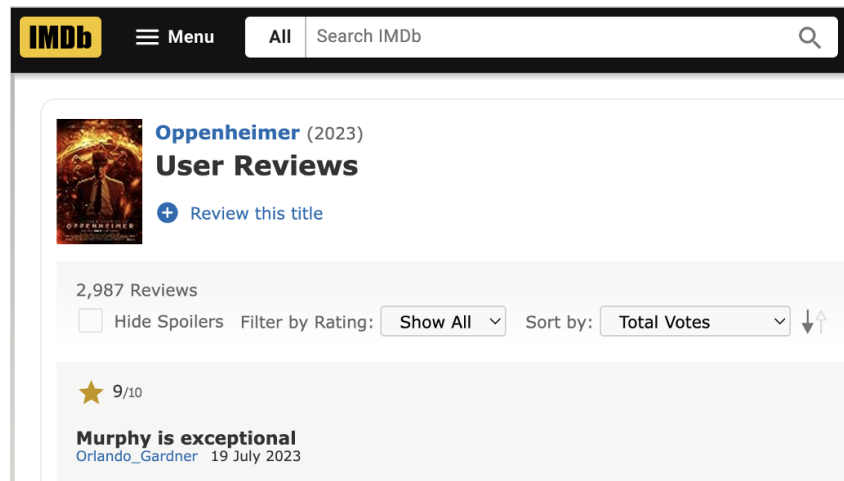# Transformers and Attention
Exercise 1.1
Advanced Deep Learning in Computer Vision

February 2025

In this exercise, you are asked to build a transformer model for sentiment classification (positive/negative) of IMDB movie reviews.

Your task is to complete the implementation of the given transformer model, train the model on the given dataset, and evaluate your results illustrating how different choices (e.g., positional embedding, number of layers, number of heads, etc, etc) impact the performance of your model at your given task.

Your tasks are as follows:

1. Complete the implementation of the attention module (`class Attention(nn.Module)`) inside the file `transformer.py`. You might find it useful to use the `"testimplementation.py"` file to test your module before training your transformer model.

2. Complete the implementation of the positional encoding (`class PositionalEncoding(nn.Module)`) inside the file `transformer.py`. You might find it useful to use the `"testimplementation.py"` file to test your module before training your transformer model.

3. Train and evaluate your model on the IMDB reviews dataset using the `textclassification.py`. With the given size of the data, it is possible to run the code locally at a CPU (1 epoch takes less than a minute).

4. Illustrate how different choices for your model affect your model's performance at your given task.

5. **OPTIONAL**: Implement a $<CLS>$ token and use it (i.e., `pool='cls'`) instead of the implemented mean option (average pooling)

6. **OPTIONAL**: Change the positional encoding with a learnable one (i.e., implement and test the `class PositionalEmbedding(nn.Module)`).

7. **OPTIONAL**: IF you want to experiment more, you can try to finetune a pre-trained BERT model on the IMDB dataset. You can find a lot of pre-trained bert models at huggingface (e.g., https://huggingface.co/distilbert/distilbert-base-uncased)

8. Your model, data, process, performance evaluation, and results should be documented and discussed in a PDF (up to 4 pages) to be uploaded on DTU Learn together with exercises 1.2 and 1.3. More details at the given template.