

E3: Multimodal Representations

Prof. Dr. Anna Rohrbach,
Prof. Dr. Marcus Rohrbach,
Hector Garcia Rodriguez, M. Sc.
Jonas Henry Grebe, M. Sc.



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Deadline Friday 4th July, 2025

General Information

In the lectures, you have learnt about [CLIP](#). In this exercise, we will explore it further, together with a more recent related approach, [SigLIP](#).

To hand in your solutions to this exercise, you will have to submit the following:

- Your report as a PDF named **E3_GROUP_<YOUR_GROUP_NUMBER>.pdf** containing your answers to the question part and any additional explanations, descriptions, or discussions required for the coding part. You will also be asked to provide code snippets for several of the coding tasks in your report. Parts of your solution that are not clearly linked to a task won't be graded.

Please use the [E3 LaTeX template](#) that we provide for this exercise. You must answer within the **solutionbox** environment, and not change the size of the boxes. You must place your textual answers using:

```
\begin{solutionbox}[N cm]
<YOUR ANSWER>
\end{solutionbox}.
```

Your answers should be precise and fit within the existing **solutionbox**. Parts of your answer outside of this box may not be taken into account. If you want to add a table, use only the **tabular** environment. **The font and any other formatting cannot be changed.**

Additionally, you must also add to the report your code snippets, using the **solutioncode** environment provided. Here's how to use it:

```
\begin{solutioncode}[language=Python]
<YOUR CODE>
\end{solutioncode}.
```

You only need to add the code in between **# ---- YOUR CODE STARTS HERE ----** and **# ---- YOUR CODE ENDS HERE ----**, since you should only edit that.

- Additionally, a [Google Colab link](#) (with edit access to everyone) to your copy of the provided notebook has to be submitted to Moodle. Ensure that the results mentioned in your report are reproducible from the submitted checkpoint of your notebook.

Important: Before you share it, you have to save the checkpoint of your notebook, which gives it a time stamp and shows it in the revision history (File → Save and checkpoint), where you can rename it to **Final Submission (Group <YOUR_GROUP_NUMBER>)**. Every change after this submission checkpoint won't be graded. If there is no pinned revision that is marked as your submission, we will just grade the last existing checkpoint from before the deadline. *If you used Kaggle, please upload your notebook to Google Colab again to provide this link.*

Any use of ChatGPT is not allowed, and it will be penalized.

In case of questions regarding the submission process, please use the **Admin Q&A** forum on Moodle.

Exercises

This exercise can achieve a maximum of 25 Points (Pts.) with up to 8 Pts. for the question and 17 Pts. for the coding part.

Question Part (8 Pts.)

1. 🍌 Describe CLIP (1 Pts.)

2. 🍌 We are using CLIP. Assume global batch size 8 and 2 GPUs. The text and image embeddings of samples 1-4 are computed in GPU 1, and samples 5-8 in GPU 2. During loss computation, assume the image embeddings do not change GPUs, and only text embeddings rotate between the GPUs. Besides the text embeddings, what tensors need to be moved between the GPUs so that each GPU holds the final loss of each of the samples whose embeddings it computed? (2 Pts.)

3. 🍌 Describe [SigLIP](#). What are the two most important reasons to use it, in comparison to CLIP? (3 Pts.)

4. 🍌 List the two most important differences between Figures 2 Left and Middle in [the SigLIP paper](#) that apply to both sigmoid and softmax versions. Provide one reason behind each difference. (2 Pts.)

Coding Part (17 Pts.)

In this section, we will revisit Imagenette, and introduce [imagenet captions](#). We will use these to finetune [SigLIP](#). The code provided in [this notebook](#) handles the dataloading, training, and evaluation. In order to finetune SigLIP with large batchsizes, you will need 2 GPUs. **Kaggle offers 30 hours weekly of 2 T4 GPUs. Use them for the coding part in this exercise. Do not forget to upload your final notebook again to colab for submission, with the cell outputs.**

1. Zero-shot Image Classification with SigLIP. (5 Pts.)



- a) 📌 Describe how SigLIP (or CLIP) can be used for image classification tasks.


- b) 📌 In the code provided, the evaluation during training is not correct. Identify and explain the error, and explain how to correct it.

- c) 📌 Correct the `train()` function based on your analysis above. The code block below computes the difference between the provided train function and your correction. Restart the notebook kernel and run all cells. Copy the output of the cell calculating the code diff into your report. In the following questions, use this corrected `train()` function. If you do not find the problem, you can continue to use the existing `train()` in the following tasks without this causing the loss of any additional points.




2. Finetuning SigLIP (7 Pts.)

- a) 📌 Run the caption-based and template-based finetuning of SigLIP. Compare the resulting classification validation accuracy. List the accuracy values and discuss the results. Why would one be better than the other?

-
- b)   For the template-based finetuning, explore the effect of batch size and number of classes on validation accuracy. Perform multiple finetuning runs for SigLIP with different values of these hyperparameters, until you find some trend between each hyperparameter and validation accuracy. Include the final validation accuracies after finetuning for each of these hyperparameter combinations in your report.

-
- c)  What correlation do you observe between each of these hyperparameters and validation accuracy? Hypothesize about the primary reasons for such results.

3. Finetuning SigLIT (5 Pts.)

- a)  Implement SigLIT. Copy your code into the report.
- b)   As done previously, explore the effect of batch size and number of classes on validation accuracy using template-based finetuning. Perform multiple finetuning runs for SigLIT with different values of these hyperparameters, until you find some trend between each hyperparameter and validation accuracy. Include the final validation accuracies after finetuning for each of these hyperparameter combinations in your report.

- c) 🖨️👉 Do you observe the same correlation between hyperparameters and accuracy as for SigLIP? Identify all the ways in which the results are different. Hypothesise about the primary reasons for such difference.