Motivation

Ever since the rapid increase in popularity and usage of AI-technologies I have found myself fascinated with both Large Language Models (LLM) and Convolutional Neural Networks (CNN), especially regarding their respective abilities and also their limitations. More specifically, I find great interest in the way they are designed to resemble the human thought process and are modelled to interact with humans, the results being mostly highly satisfying and occasionally comedic. Albeit on a small scale, this project aims to analyze whether a model, trained on a specific dataset, can recognize distinct features and transfer it's knowledge onto a separate unique dataset to perform accurate predictions the way a human would. I have taken inspiration from various Huggingface libraries to tackle these tasks.

In a first step, an image classification model, trained on a diverse dataset of mostly human images categorized by distinct emotional expressions - happiness, sadness, anger, and fatigue – is created. Following this, samples from two separate image datasets are taken to yield predictions on the respective emotions. The first one will show human expression of sentiments once more, the second one however will show emotions of dogs, thereby exploring cross-species emotional recognition. The average prediction of each sample for the correct emotion will be calculated and compared to one another in a final step.

On a broader scale, while this project likely won't produce ground-breaking impact, it can serve as initialization on continuous improvement in both human-computer interactions as well as in fields of behavioural studies.

Training / Fine Tuning

For the training and fine tuning I experimented around with numerous different inputs. The results, for the most part, certainly left much to be desired. The loss as well as the accuracy both seem to be subject to overfitting in this model. I image the main reason being that the dataset is only limited to 150 images per term, which will often result in the training performance being inaccurate. I ran the testing on 5 epochs, as this saved significant time. The plan was to run on 10 epochs in the final version.
Initially the batch size in the loaders was set to 8, as seen in the image below:

Initialize Transformer Feature Extractor and Image Classifier.

```
feature_extractor = ViTFeatureExtractor.from_pretrained('google/vit-base-patch16-224-in21k')
model = ViTForImageClassification.from_pretrained(
    'google/vit-base-patch16-224-in21k',
    num_labels=len(label2id),
    label2id=label2id,
    id2label=id2label
)
collator = ImageClassificationCollator(feature_extractor)
train_loader = DataLoader(train_ds, batch_size=8, collate_fn=collator, num_workers=2, shuffle=True)
val_loader = DataLoader(val_ds, batch_size=8, collate_fn=collator, num_workers=2)
```

Figure 1 - Batch size in training loader

This produced the following mildly satisfying output in the subplots:

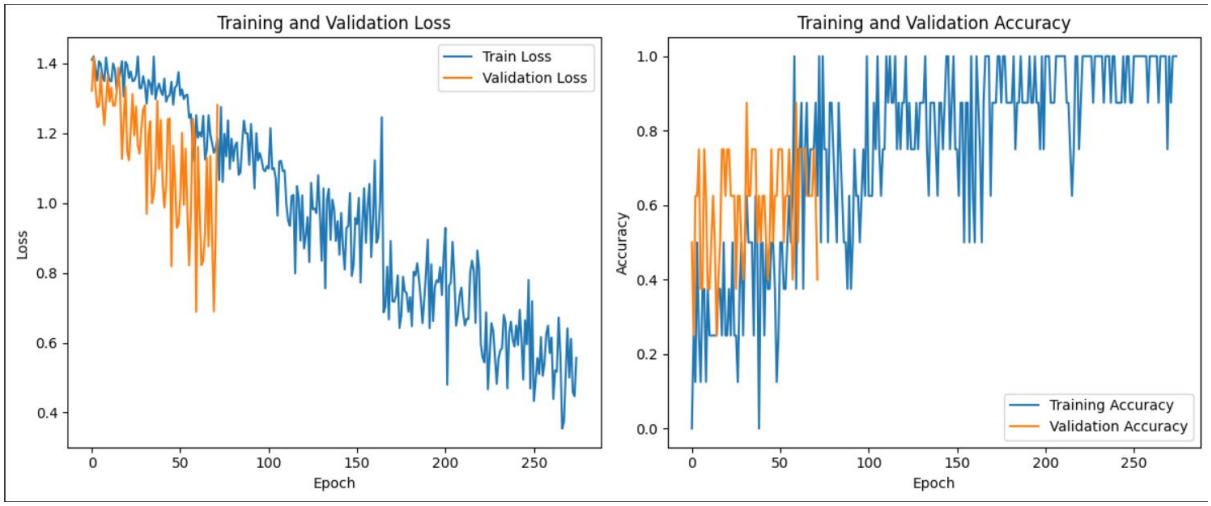

Figure 2 - Results with batch size 8

Trying to improve the fine tuning, I altered the batch size to 16. This seemed to have a slightly positive impact on the overfitting problem. For the loss-function, however, the validation accuracy was rather bemusing:

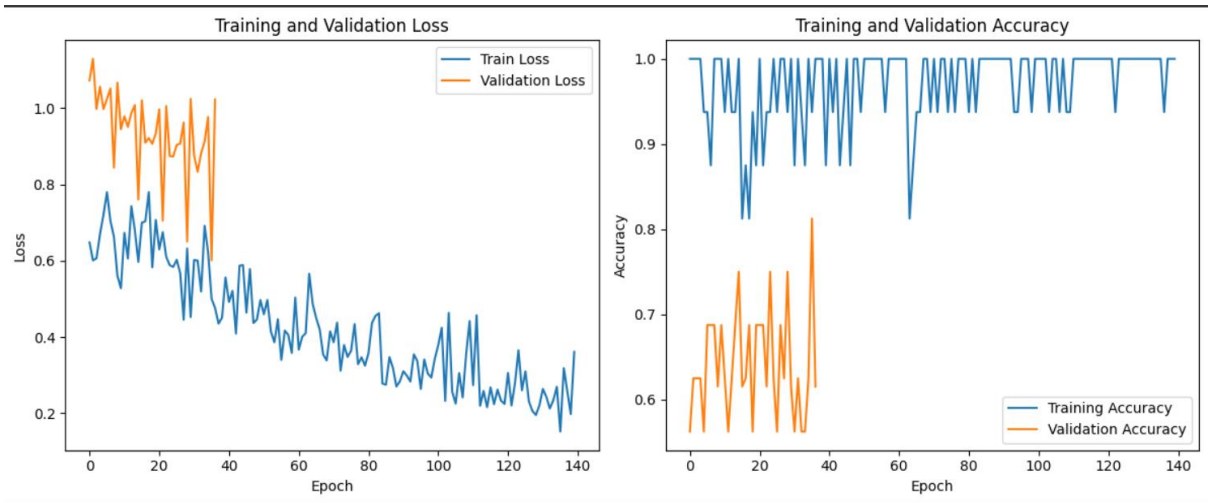

Figure 3 - Results with batch size 16

The logical follow up presented itself in reducing the batch size to 4, which increased the variance even more:

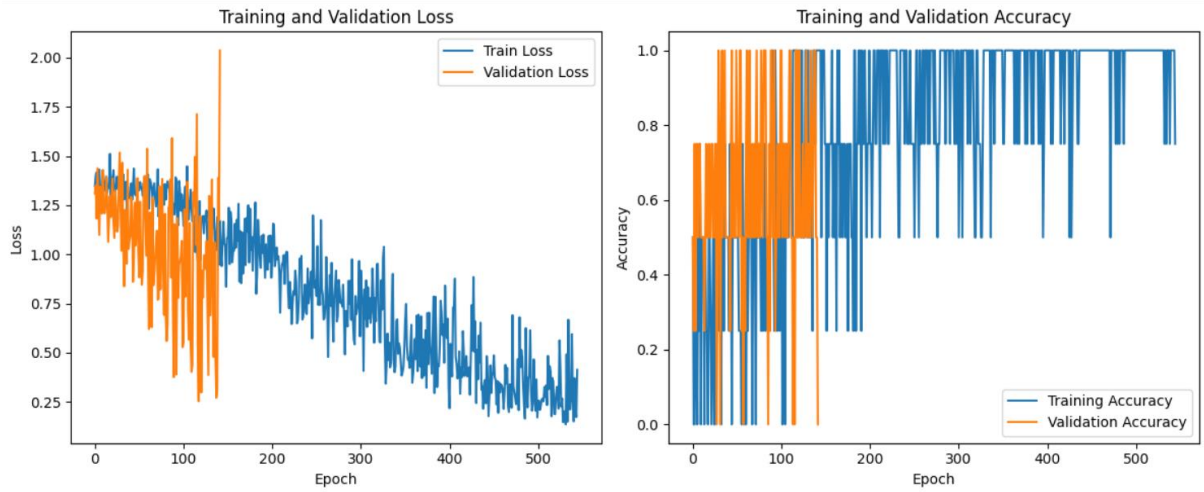

Figure 4 - Results with batch size 4

Finally, I decided on setting the batch size to 24 and set the max epochs to 8. This ended up producing a significantly more satisfying result in both instances, with all the graphs showing some stability and decreased variance. No doubt both sets are overfitted, however considering the limited data I found this acceptable. I refrained from increasing the batch size, as this might lead to less accurate prediction outputs in the eventual classification model.

```
pl.seed_everything(42)
classifier = Classifier(model, lr=2e-5)
trainer = pl.Trainer(accelerator='gpu', devices=1, precision=16, max_epochs=8)
trainer.fit(classifier, train_loader, val_loader)
```

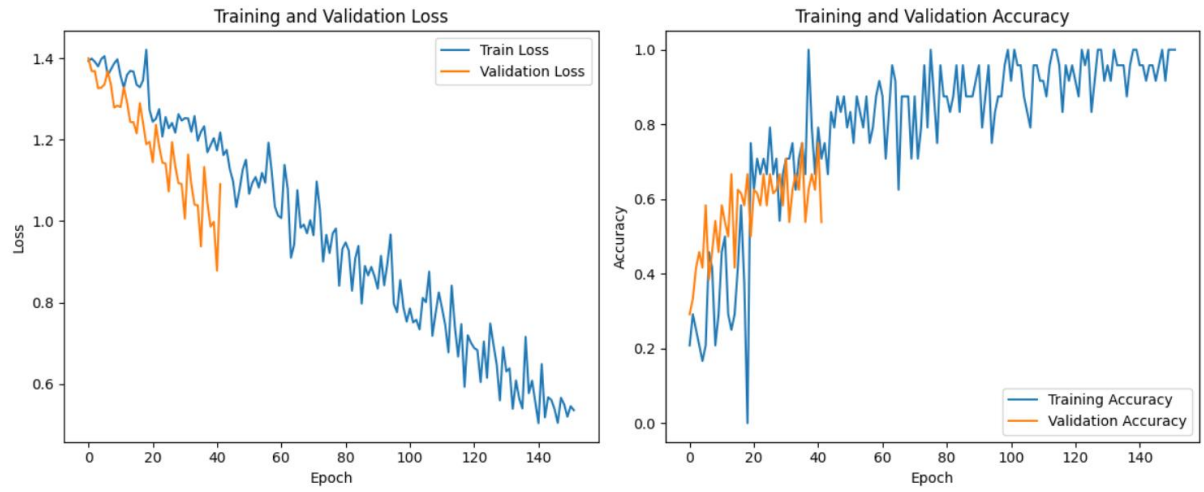

Figure 5 -Results with batch size 24, epochs 8

Interpretation & Validation

For this step I decided to gather two datasets from Kaggle, the first one on an image collection from human emotions and the second one on dog emotions. I performed some data cleansing and then let the model make predictions on 20 images of each category.

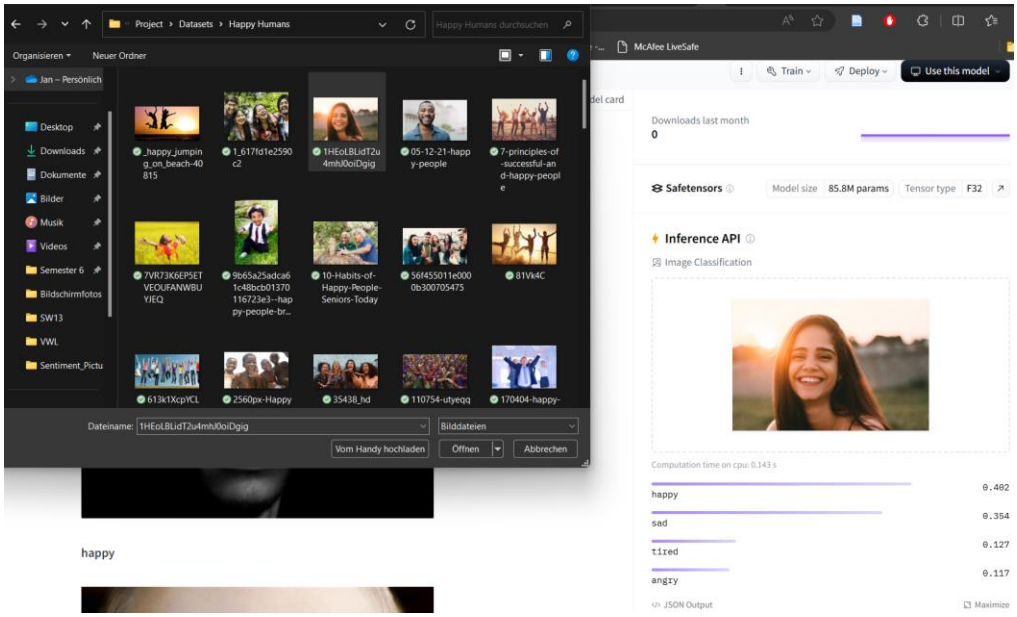

Figure 6 - Prediction on human emotion

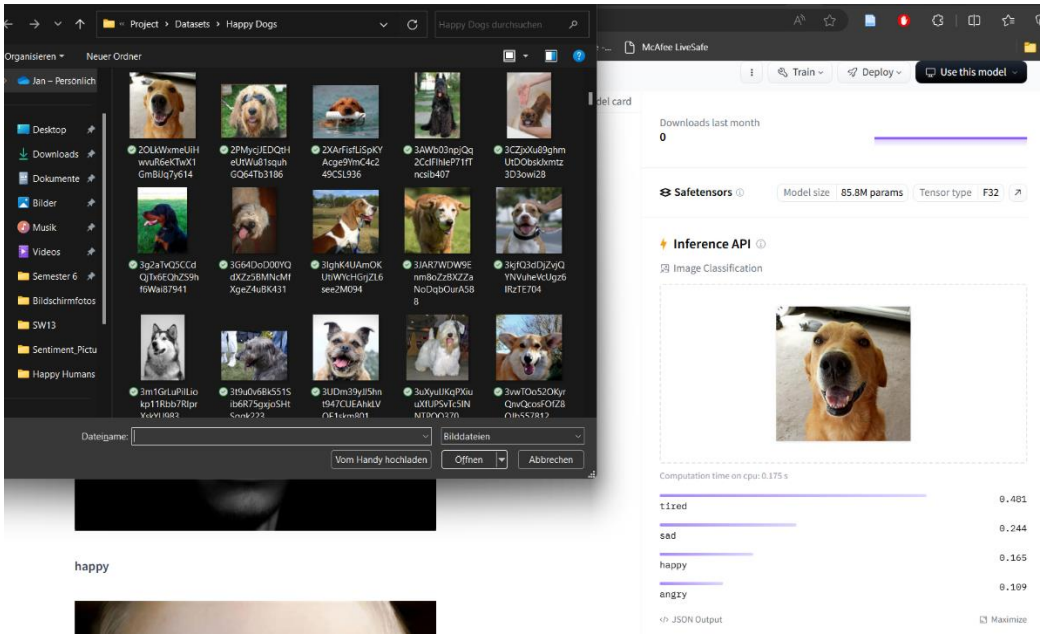

Figure 7 - Prediction on dog emotion

In both cases I used images from the “happy” dataset to generate a true comparison. Across the board the model does not exude great confidence in any case. For the human side it did predict the correct emotion in 19 out of 20 cases, with an average accuracy score of 39.5%. For the predictions on happy dogs the results looked rather dire, as expected. In 20 out of 20

cases it predicted “tired” as the most likely emotion, which was an interesting find. It did predict “happy” as the second most likely sentiment in 18 cases, resulting in an average accuracy score of 22.7%.

This are just two examples of the dataset from which we can work with, a more complete set can be found here.

Reflection / Limitations

Overall, it can be stated the finalized version this project shows both promise and room for improvement. It is satisfactory that the created image classification model is functional in it’s core. However, the results in the training process as well as in the eventual execution are far from perfection. As previously mentioned, this will mainly be the cause of the rather small initial dataset, which unfortunately did not seem able to being increased. Having already initiated the project and run various tests, I refrained from changing the data source and calling from a different API, as this might have put the functionality of the project at risk.

Nevertheless, I certainly profited in the way of increase of both skill and knowledge in the topic of classification models. And if nothing else, as mentioned in the motivation chapter, this project can undoubtably serve as a foundation to be built upon and inspire miscellaneous fields of research.