

Streamlit Image Classification

⌚ Status	In progress
👥 Assign	

Link → <https://github.com/pytholic/streamlit-image-classification>

Overview

The main goal of the project is to build an image classification pipeline and deploy the model using Streamlit.

Notes

2023.03.02

- To run the app

```
streamlit run app/app.py
```



- Bar styling with css

```
# Show progress bar with probabilities
st.markdown(
    """
<style>
.stProgress .st-b8 {
    background-color: orange;
}
</style>
```

```
        """
        unsafe_allow_html=True,
    )
st.progress(prob)
```

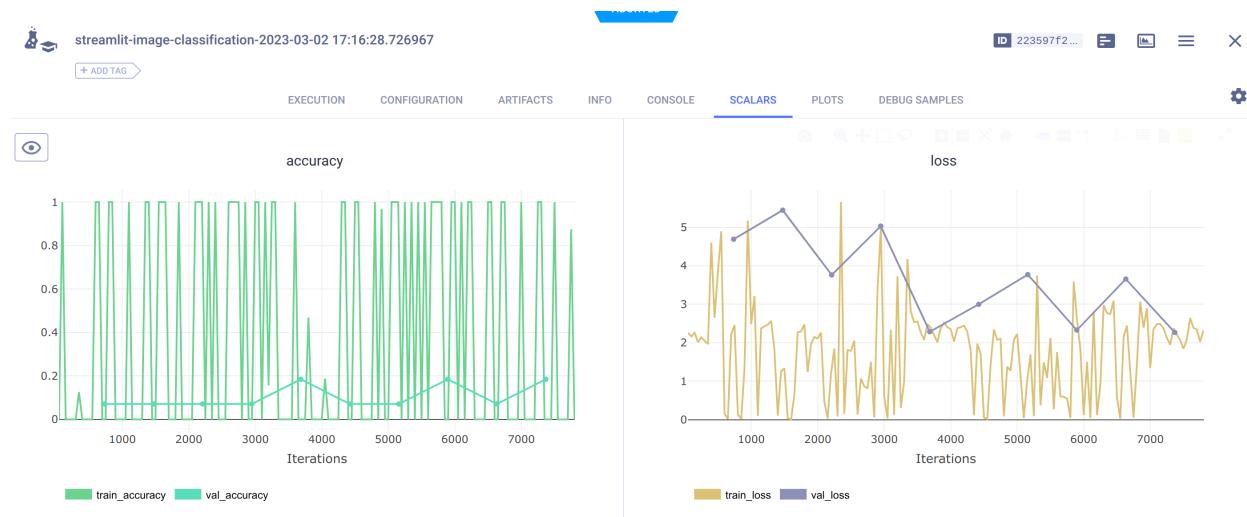
- Need to train model on full data and deploy on hugging face

2023.03.03

Streamlit Demo

Training model on full data.

Model is not learning



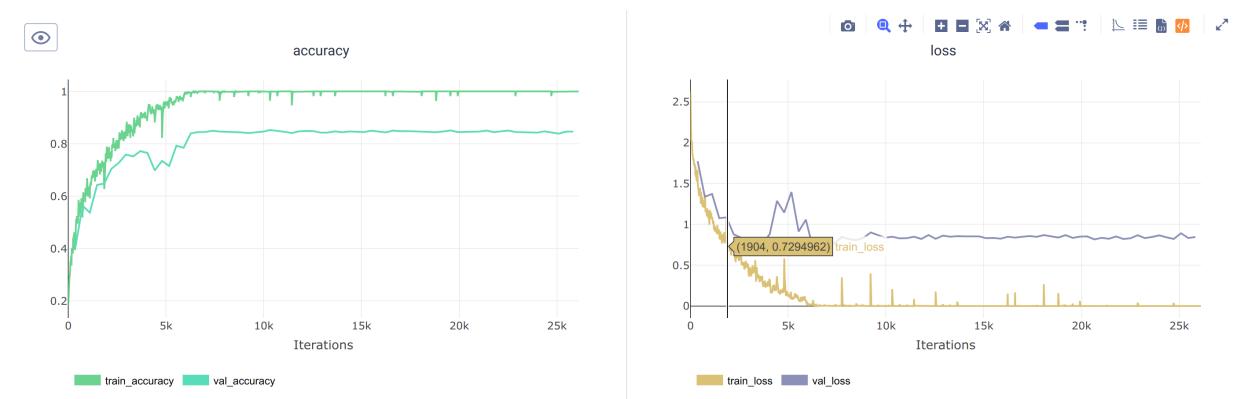
Gonna use pre-trained model now for faster convergence.

Try fixes:

- Pretrained false
- Optimizer = Adam
- Change loss
- Different model
- Changing transformations order

It was because I missed `shuffle` in dataloaders.

Using `ResNet-18` instead. Results are fine.



Trained with `ce_loss` and `nll` loss. There was some probability issue with `nll` loss, I already applied log in the model, so I was trying to apply it again which was issue.

```
# Prediction
    # Make a prediction on the image
    with torch.no_grad():
        output = model(image)
        # convert to probabilities
        probabilities = torch.nn.functional.softmax(torch.exp(output[0], dim=1))

        topk_prob, topk_label = torch.topk(probabilities, 3)
```

Anyway, I switch to `ce_loss` and using logits now.

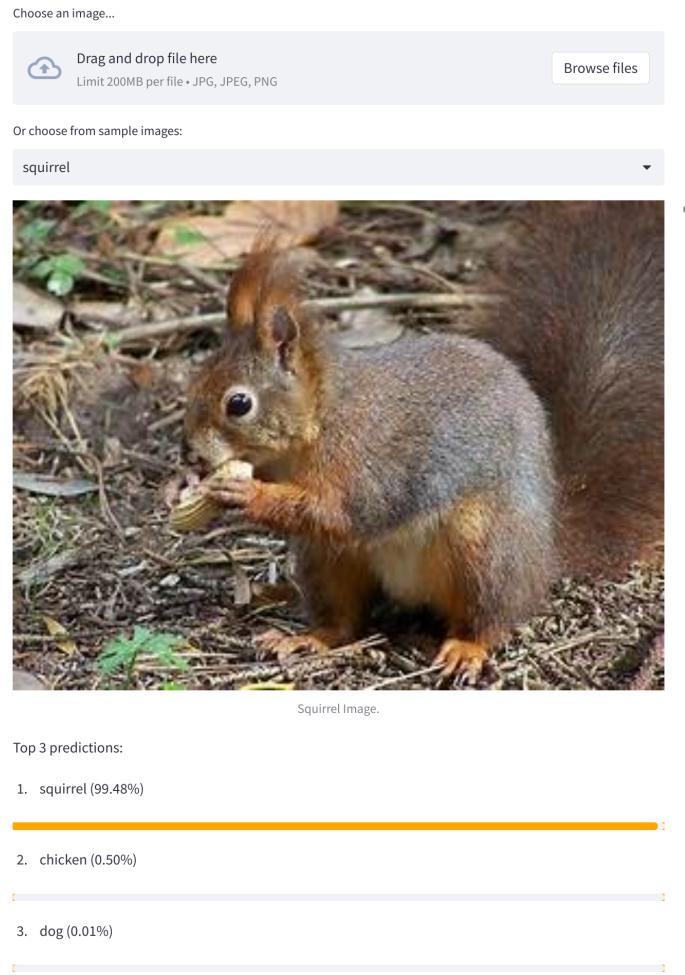
```
# Prediction
    # Make a prediction on the image
    with torch.no_grad():
        output = model(image)
        # convert to probabilities
        probabilities = torch.nn.functional.softmax(output[0])

        topk_prob, topk_label = torch.topk(probabilities, 3)
```

So finally I have a working ResNet-18 model.

We can run the demo locally.

Animal-10 Image Classification



Deployment

Deploying on hugging faces.

Create a new Streamlit space.

Clone the repository. Do not remove `README.md`, it contain configuration.

Add you `app.py` and `requirements.txt`. Then push. It will build and deploy.

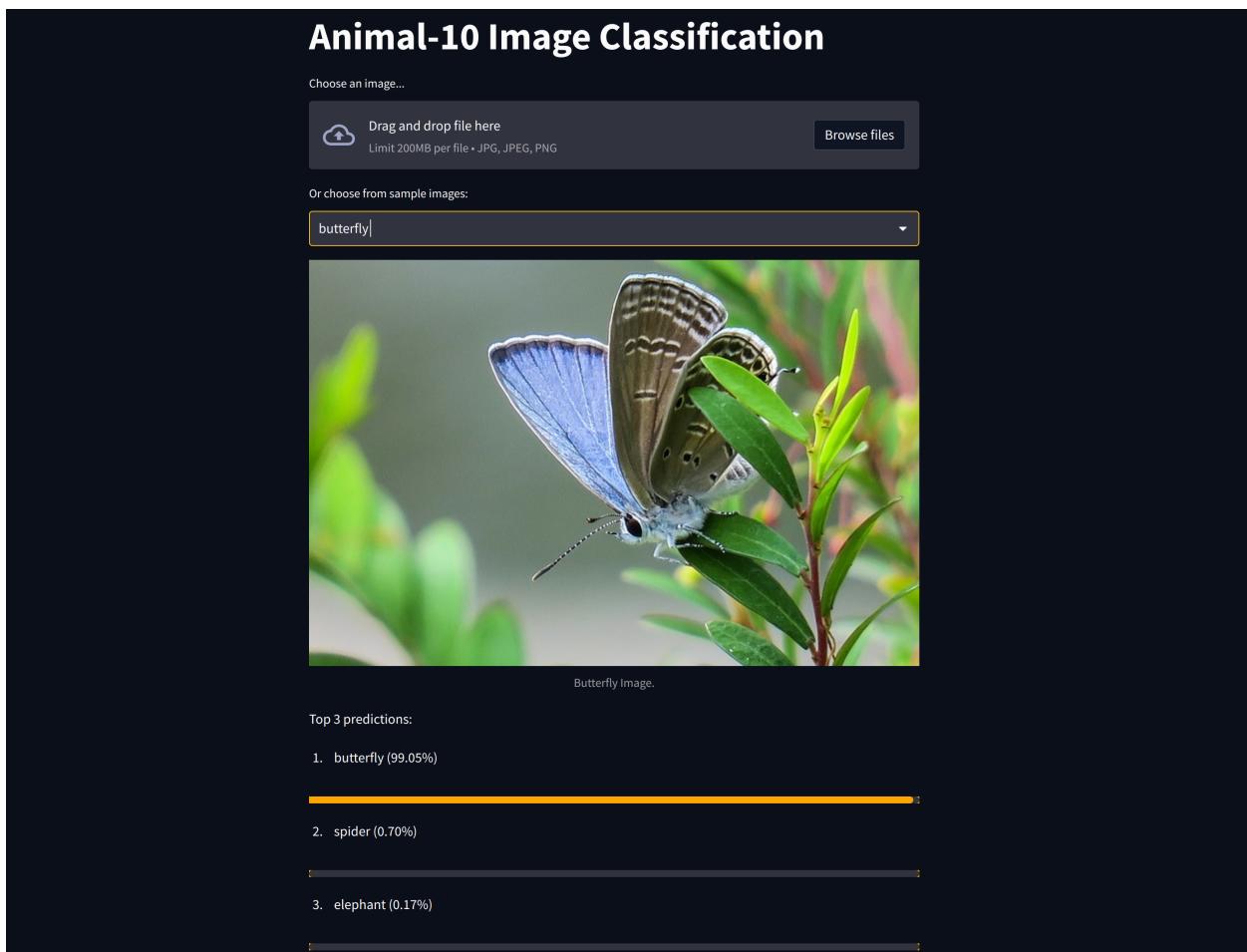
We will push other files and `model` later.

Then we add all other files like `model.py`, `config`, images, model weights etc.

Also update `requirements.txt` if required.

Might have to add some files to git lfs.

Demo is live now.



Links

Transfer learning

<https://wandb.ai/wandb/wandb-lightning/reports/Transfer-Learning-Using-PyTorch-Lightning--VmIldzoyODk2MjA>

Probabilites of predictions

<https://pytorch.org/docs/stable/generated/torch.topk.html>

Getting started with repo

<https://huggingface.co/docs/hub/repositories-getting-started#terminal>