Project - Evaluation

Davide Pintus
2074672

Setup

Originally, 3 types of networks (ResNet, U-Net, ViT) were trained for the experiment, each with 5 possible lead times (6, 24, 72, 120, 240 [hours]). For this training, 50 epochs were used, training with data from 1979 to 2015; these data were divided into batches with a batch size of 128. For the validation set, data from 2016 were used, while data from 2017 and 2018 were used for testing.
The hyperparameters of the 3 network types are shown below.

ResNet		U-Net		ViT	
Padding size	1	Padding size	1	Patch size	2
Kernel size	3	Kernel size	3	Embedding dimension	128
Stride	1	Stride	1	# ViT blocks	8
Hidden dimension	128	Hidden dimension	64	# heads	4
Residual blocks	28	Channel multiplications	[1, 2, 2]	MLP ratio	4
Dropout	0.1	# blocks	2	Prediction depth	2
		Dropout	0.1	Hidden dimension	128
				Drop path	0.1
				Dropout	0.1

Having very limited computing power, I had to reduce the amount of data used for train, validation and test sets: for train I used data from 1980, 1981 and 1982, while for validation and test I split the 1986 data into two equal parts, allocating one half for each. It was also necessary to reduce the number of training epochs, using only one as test of the trainig. The complexity of the networks was also decreased, using the following hyperparameters:

ResNet		U-Net		ViT	
Padding size	1	Padding size	1	Patch size	2
Kernel size	3	Kernel size	3	Embedding dimension	32
Stride	1	Stride	1	# ViT blocks	2
Hidden dimension	32	Hidden dimension	16	# heads	2
Residual blocks	4	Channel multiplications	[1, 2, 2]	MLP ratio	4
Dropout	0.1	# blocks	1	Prediction depth	1
		Dropout	0.1	Hidden dimension	32
				Drop path	0.1
				Dropout	0.1

Results

The two metrics proposed by the paper were used to evaluate the performance of the trained models: the latitude-weighted root mean squared error (RMSE) and the anomaly correlation coefficient (ACC).

$$RMSE = \frac{1}{N}\sum_{k=1}^{N}\sqrt{\frac{1}{H\cdot W}\sum_{i=1}^{H}\sum_{j=1}^{W} L(i)\cdot(\tilde{X}_{k,i,j} - X_{k,i,j})^2}$$

$$ACC = \frac{\sum_{k,i,j} L(i)\cdot \tilde{X}'_{k,i,j}\cdot X'_{k,i,j}}{\sqrt{\left[\sum_{k,i,j} L(i)(\tilde{X}'_{k,i,j})^2\right]\cdot\left[\sum_{k,i,j} L(i)(X'_{k,i,j})^2\right]}}$$

$$\tilde{X}' = \tilde{X}' - C \qquad X' = X' - C$$

where:

- N : number of datapoints;
- H : number of latitude coordinates;
- W : number of longitude coordinates;
- X and $\tilde{X}$: ground-truth and prediction;
- $L(i)$: latitue weighting function;
- C : temporal mean of the ground truth data over the entire test set.

$$L(i) = \frac{\cos(H_i)}{\frac{1}{H}\sum_{j=1}^{H}\cos(H_j)} \qquad C = \frac{1}{N}\sum_{i=1}^{N} X_i$$

Below are graphs of the performance of the trained models on the 3 predicted variables:

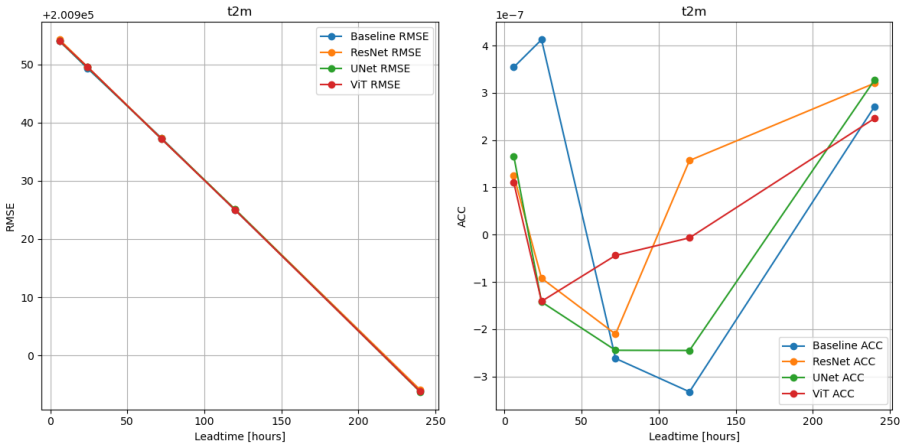

Figure 1: Performance on forecasting T2m at different lead times

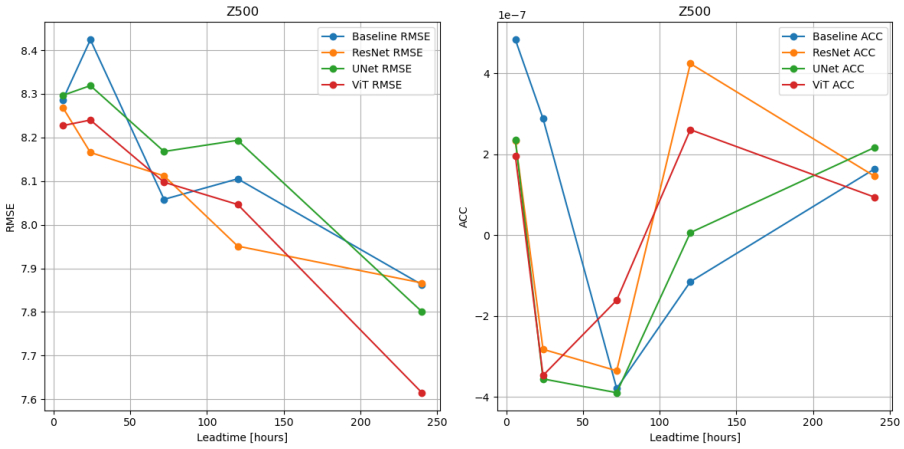

Figure 2: Performance on forecasting Z500 at different lead times

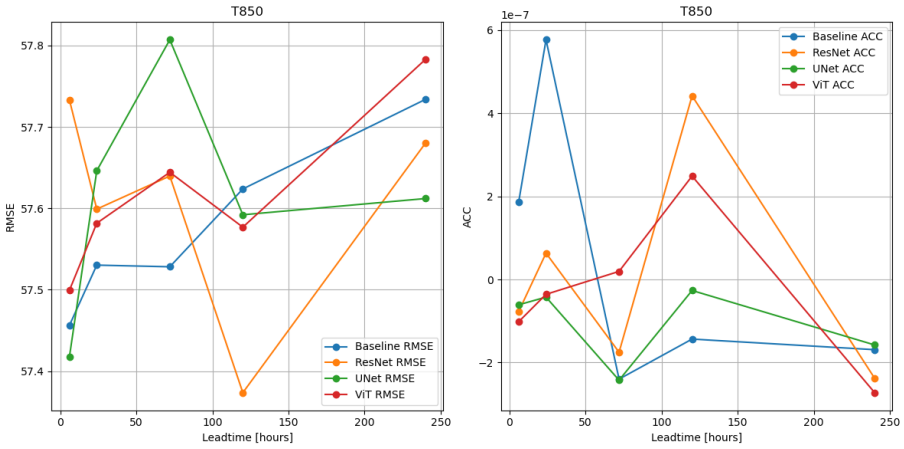

Figure 3: Performance on forecasting T850 at different lead times

Due to the low computing power at my disposal, the training was not sufficient to achieve good results. With good training, one expects that the RMSE, whose ideal value should be as low as possible, tends to increase with increasing lead time; on the other hand, one expects that the ACC, whose ideal value should be as high as possible, tends to decrease with increasing lead time.