Project - Description

Davide Pintus
2074672

Based on the 'ClimateLearn: Benchmarking Machine Learning for Weather and Climate Modeling' paper, I re-implemented the models used for the 'weather forecasting' task. For this task, models are trained to predict future values (lead times, 6, 24, 72, 120, 240 [hours]) of certain weather variables from information on the current weather and the weather of the previous 6 and 12 hours; to do it, it is used a supervised learning approach.
For the experiment, it was used the ERA5 dataset at 5.625°, which contains three types of variables: constants, surface variables and atmospheric properties at the chosen altitudes; each variable is a matrix of size 32x64 and is standardised (mean 0, standard deviation 1) before being given as input to a model.

Type	Variable name	Abbrev.	Levels
Static	Land-sea mask	LSM	
Static	Orography		
Static	Latitude		
Single	Toa incident solar radiation	Tisr	
Single	2 metre temperature	T2m	
Single	10 metre U wind component	U10	
Single	10 metre V wind component	V10	
Atmospheric	Geopotential	Z	50, 250, 500, 600, 700, 850, 925
Atmospheric	U wind component	U	50, 250, 500, 600, 700, 850, 925
Atmospheric	V wind component	V	50, 250, 500, 600, 700, 850, 925
Atmospheric	Temperature	T	50, 250, 500, 600, 700, 850, 925
Atmospheric	Specific humidity	Q	50, 250, 500, 600, 700, 850, 925
Atmospheric	Relative humidity	R	50, 250, 500, 600, 700, 850, 925

Figure 1: ERA5 Variables

During training, AdamW optimiser is used to adjust the networks, using a linear warmup schedule for the first 5 epochs and cosine-annealing for all the others; early stopping with patience of 5 epochs is also used. The loss function used for training is the latitude-weighted mean squared error function. For evaluation, the latitude-weighted root mean squared error (RMSE) and anomaly correlation coefficient (ACC) are used as metrics.
The models used are of three types: ResNet, UNet and ViT.
The ResNet model is based on the use of a shortcut connection that allows it

to learn from the sum of the original input and the output from a deeper layer. Here an example:

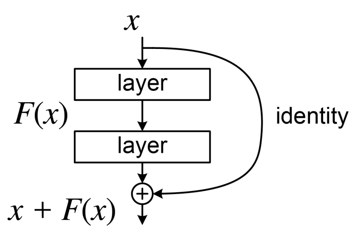

Figure 2: ResNet Block

The U-Net consists of two phases: a contracting phase and an expansive phase. The contracting phase layers capture specific feature and reduce the spatial resolution of the input, while the expansive phase layers decode the encoded data and use the information from the contracting path via skip connections to generate the output. Here an example:

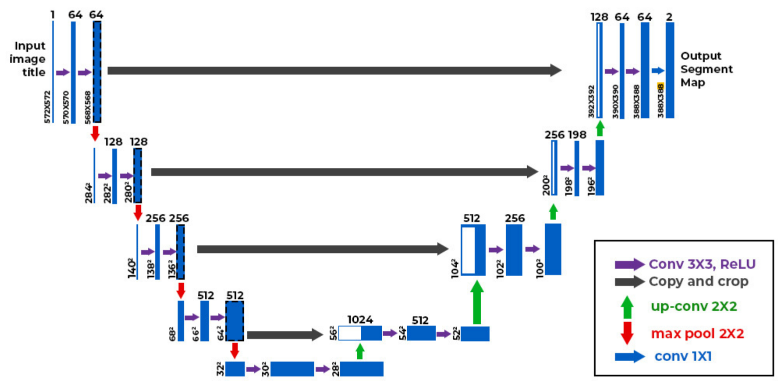

Figure 3: U-Net

The ViT is a model based on the Transformer architecture. Such blocks are used to process the input by dividing it into patches, analyzing the relationships between these patches, and ultimately classifying the input. Here an example:

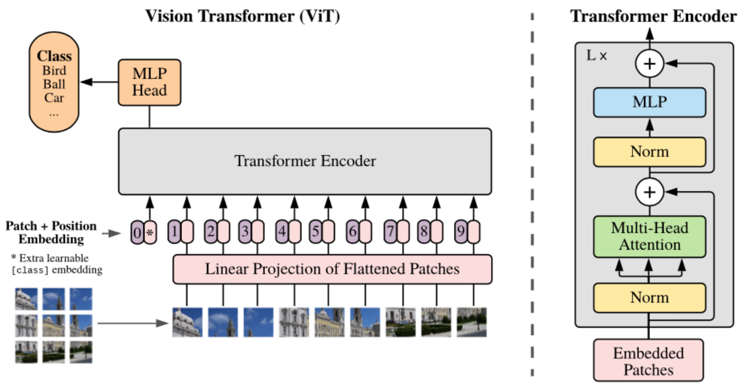

Figure 4: ViT

For the evaluation, I implemented a simple baseline based on 2D convolution. Such baseline was trained together with the other models.